Reflective Cross-Granularity Grounding with Preference Optimization for Long Video Understanding

Wei Feng
DCST, Tsinghua University
Beijing, China
fw22@mails.tsinghua.edu.cn

Xin Wang*
DCST, BNRist, Tsinghua University
Beijing, China
xin_wang@tsinghua.edu.cn

Hong Chen
DCST, Tsinghua University
Beijing, China
h-chen20@mails.tsinghua.edu.cn

Yu-Wei Zhan
DCST, Tsinghua University
Beijing, China
zhanyuweilif@gmail.com

Zihan Song
DCST, Tsinghua University
Beijing, China
songzh23@mails.tsinghua.edu.cn

Bin Huang
DCST, Tsinghua University
Beijing, China
huangb23@mails.tsinghua.edu.cn

Kecheng Zheng
Ant Research, Hangzhou, China
zkechengzk@gmail.com

Wenwu Zhu*
DCST, BNRist, Tsinghua University
Beijing, China
wwzhu@tsinghua.edu.cn

Abstract

Video Large Language Models (video LLMs) have demonstrated remarkable capabilities in video understanding tasks, such as video question answering and temporal localization. However, understanding long videos still remains a significant challenge. Existing video LLMs adopt uni-granularity tokens for long videos, failing to simultaneously understand both high-level semantics and low-level visual details in videos. To tackle this problem, we propose **ReCrossVLLM**, a reflective cross-granularity grounding framework for video LLM with preference optimization to collaboratively achieve long video understanding, which not only retains the capabilities of high-level video semantics understanding, but also strengthens the fine-grained understanding abilities. Specifically, we propose the coarse-to-fine grounding and fine-to-coarse reflection strategies for long video understanding. In the coarse-to-fine grounding strategy, the video LLM with a coarse-grained module first locates the key video segments from the long video by tackling massive frames of the long video with fewer per-frame tokens. And then video LLM adapted with the fine-grained module further analyzes the key video segments with more per-frame tokens so that it can understand fine-grained information. In case the video LLM locates the wrong key video segments, during the inference stage, our designed fine-to-coarse reflection strategy instructs the fine-grained module to reflect the effectiveness of the locating result and decide whether to return to the coarse-to-fine grounding strategy with reflection feedback. Additionally, during the training stage, the coarse-to-fine grounding strategy is optimized with our proposed cross-granularity preference optimization strategy to further improve grounding efficiency. Extensive experiments for long video question answering and temporal video grounding tasks demonstrate that our proposed ReCrossVLLM framework can significantly improve the Video Large Language Model for long video understanding.

CCS Concepts

• **Information systems** → *Multimedia information systems*; *Retrieval tasks and goals*; **Question answering**; • **Computing methodologies** → **Computer vision**.

Keywords

Video Large Language Model, Long Video Understanding, Video Grounding

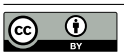
ACM Reference Format:
Wei Feng, Xin Wang, Hong Chen, Yu-Wei Zhan, Zihan Song, Bin Huang, Kecheng Zheng, and Wenwu Zhu. 2026. Reflective Cross-Granularity Grounding with Preference Optimization for Long Video Understanding. In *International Conference on Multimedia Retrieval (ICMR '26), June 16–19, 2026, Amsterdam, Netherlands.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3805622.3810717

1 Introduction

Recently, Video Large Language Models (video LLMs) have made significant progress in video understanding tasks such as video question answering and temporal localization [48]. By leveraging techniques like modality alignment and visual instruction tuning, several models [15, 18, 29] have been developed to improve temporal video representation learning and comprehension.

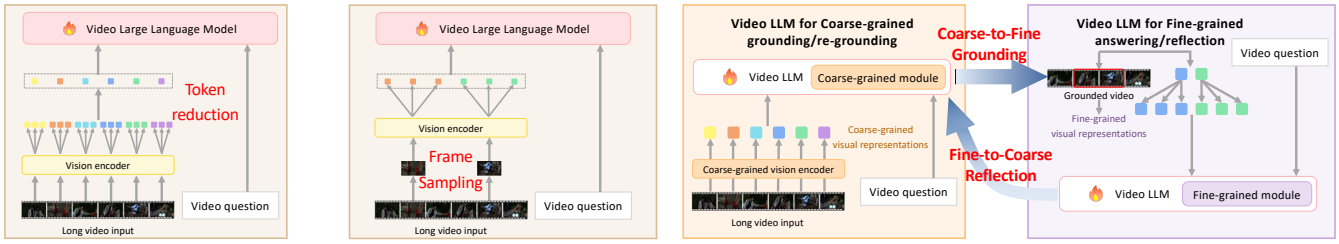
Despite the success of video LLMs, understanding long videos still remains a significant challenge [34, 42, 52]. Compared to short videos, long videos involve much more frames and thus require much more tokens that may exceed the token length limits of the LLMs [31, 41]. Existing video LLMs for long videos generally adopt uni-granularity tokens to represent the videos and then adopt the token pruning methods or frame sampling methods to reduce the token number [1, 15, 33]. However, the token pruning methods may

*Corresponding authors.



ICMR '26, Amsterdam, Netherlands

ACM ISBN 979-8-4007-2617-0/2026/06
https://doi.org/10.1145/3805622.3810717



(a) Existing Video LLMs with uni-granularity video tokens through token reduction

(b) Existing Video LLMs with uni-granularity video tokens through frame sampling

(c) Our reflective cross-granularity grounding with preference optimization for long video understanding

Figure 1: Comparison between existing works and our cross-granularity video LLM framework. Figure 1(a) denotes token reduction for long video understanding, which limits the video LLMs for detailed video perception. Figure 1(b) represents the frame sampling method, which would ignore massive intermediate question-related frames when analyzing long videos.

suffer from losing visual details, and the frame sampling methods will even lose high-level temporal semantic context.

To address these problems, as shown in Figure 1, different from existing works that adopt uni-granularity tokens and then reduce tokens, we propose ReCrossVLLM, a novel reflective cross-granularity grounding framework for video LLM, with adaptive modules of different granularities and cross-granularity preference optimization to collaboratively achieve long video understanding. The proposed **ReCrossVLLM** works similarly to the way that humans try to answer a question for a given long video, where we first skim through the video to roughly understand the global semantics of the video, and then watch the segments relevant to the question in a fine-grained manner. Specifically, ReCrossVLLM consists of: i) a Video LLM with a coarse-grained module that processes more video frames with fewer tokens per frame, tackling massive frames of the long video; and and ii) a video LLM with a fine-grained module processing fewer frames through more per-frame visual tokens. With the cross-granularity modules, we design coarse-to-fine grounding and fine-to-coarse reflection strategies for cross-granularity long video understanding. During the coarse-to-fine grounding strategies, we first utilize the video LLM with the coarse-grained module to locate the key video segments from the long video according to the textual input. And then instruct the video LLM adapted with the fine-grained module to further analyze the key video segments with more per-frame tokens so that it can understand fine-grained visual information. In case the video LLM locates the wrong key video segments, during the inference stage of our cross-granularity method, we design the fine-to-coarse strategy, prompting the video LLM with the fine-grained module to reflect the effectiveness of the locating result, and decide whether to return information to the coarse-to-fine grounding strategy with prior reflection. In addition, during the training stage, we introduce a cross-granularity preference optimization strategy (based on direct preference optimization, DPO [28]) to further improve grounding efficiency by learning from pairwise preferences among candidate policies. Extensive experiments demonstrate that our ReCrossVLLM framework significantly outperforms existing Video LLMs in long video question answering and temporal video grounding tasks.

To summarize, we make the following contributions:

- We propose ReCrossVLLM, a reflective cross-granularity grounding framework with preference optimization for long video understanding.
- We propose coarse-to-fine grounding and fine-to-coarse reflection strategies, where the grounding process is further optimized via cross-granularity preference optimization.
- Extensive experiments show that ReCrossVLLM outperforms state-of-the-art baselines in long video question answering and temporal grounding, demonstrating its superiority in capturing both high-level semantics and low-level visual details.

2 Related Work

2.1 Video Large Language Model

With the rapid development of Large Language Models [36], significant research has been devoted to enabling LLMs for temporal visual information understanding [18, 48]. These Large Language Models capable of processing video input, could be collectively referred to as Video Large Language Models. To process multimodal information, normally video LLMs first would employ large-scale multimodal datasets [4, 22, 30] with images, and videos paired with textual input to align visual features with the feature space of LLMs through the visual encoders, and use an amount of GPT-annotated or human-annotated datasets for instruction tuning in the second stage [23]. Many video LLMs, such as Video-LLaMA [7, 48], VideoChat [18], and Video-LLaVA [21] share similar methods with image LLMs [1, 5] through inputting entire image patches or massive visual tokens (with more than 200 tokens per frame) to the transformer architecture [40] of LLMs. These methods, however, all have critical challenges when dealing with long video understanding tasks due to exceeding the maximum token limitations for visual representation or only processing a small number of sampled video frames in the video, resulting in the loss of keyframe capture.

2.2 Video LLMs for Long Video Understanding

To address the critical challenges of long video understanding, video LLMs such as LLaMA-VID [20], MovieChat [33], and LongVU [31] have been developed for long video understanding through compressing each video frame into one or a few visual tokens so that the language model is able to handle all the visual representations from the entire video input [32]. However, since these methods are pre-trained from a short video understanding model and reduced visual representation features of each video frame, they are unable to analyze detailed visual information for long videos.

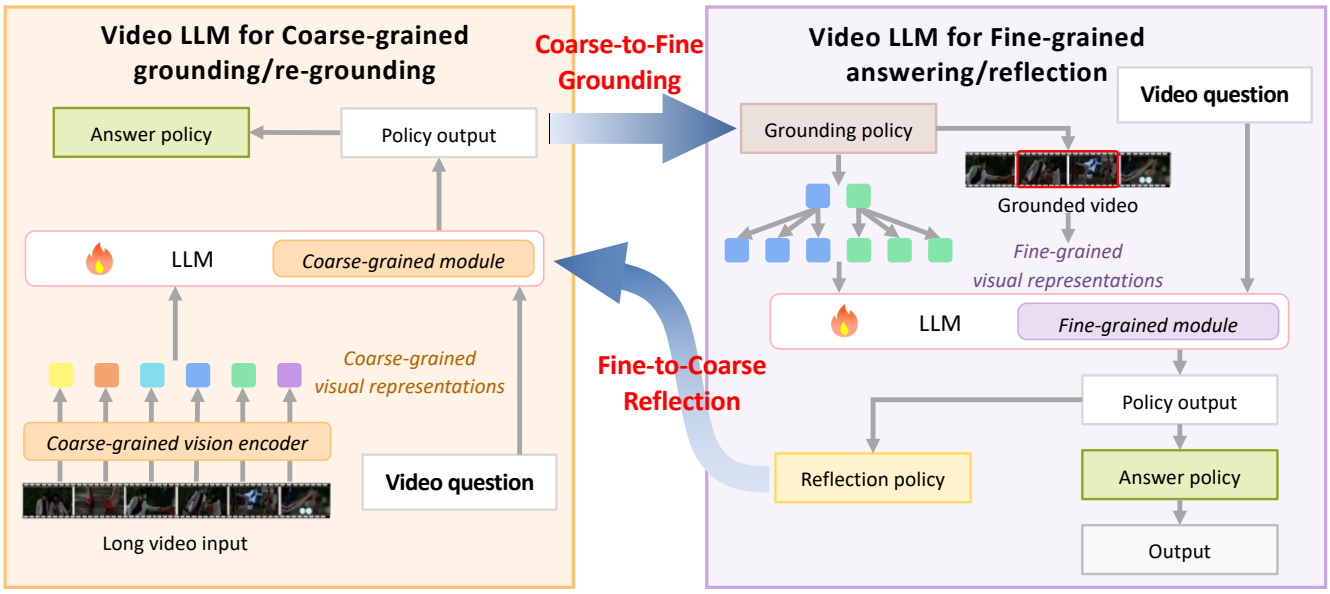


Figure 2: Our ReCrossVLLM framework, including a video LLM with a coarse-grained module to process more visual frames, and a video LLM with a fine-grained module processing more per-frame visual tokens of fewer frames. In our designed coarse-to-fine grounding strategy, the coarse-grained video LLM would localize the question-related short video segment and send it to the fine-grained video LLM for detailed perception. In case the coarse-grained video LLM locates the wrong video segments, during our fine-to-coarse reflection strategy, the fine-grained video LLM will reflect the effectiveness of the grounding result and determine whether to go back to the coarse-to-fine grounding strategy for re-grounding with prior feedback.

On the other hand, some methods, such as VideoTree [39] and VideoAgent [38], adopted the paradigm of first selecting keyframes for next-stage detailed perception [35], aiming to expand two-stage strategies [45, 47] to long video understanding. However, these methods primarily employ training-free approaches or only involve basic model fine-tuning, failing to optimize the overall strategy. They overlook the reflection refinement and specific training for strategies when the model incorrectly selects keyframes, which would compromise the accuracy of long video analysis.

3 The Proposed ReCrossVLLM Framework

In this section, we will introduce our ReCrossVLLM framework. As shown in Figure 2, our ReCrossVLLM framework includes video LLMs with a coarse-grained module and a fine-grained module, collaborating through our designed coarse-to-fine grounding and fine-to-coarse reflection strategies. In the coarse-to-fine grounding strategy, video LLM processes a large number of frames from the long video using fewer visual tokens per frame, allowing the coarse-grained module to efficiently identify key video segments. The key segment is then passed to the fine-grained video LLM, which analyzes it in greater detail using more tokens per frame to capture fine-grained visual information. In the fine-to-coarse reflection strategy, in case the coarse-grained video LLM locates the wrong key video segments, the fine-grained video LLM will reflect the effectiveness of the grounding result and determine whether to go back to the coarse-to-fine grounding strategy for re-grounding correct segments.

3.1 Coarse-to-Fine Grounding

The coarse-to-fine grounding includes a coarse-grained grounding model and a fine-grained answering model, which are our pre-finetuned coarse-grained video LLM and fine-grained video LLM, respectively.

Coarse-grained Grounding. Given a long video input $v \in R^{T\times H\times W\times C}$ with T frames, the coarse-grained video LLM would uniformly sample N frames, represented as $\widetilde{v} \in R^{N\times H\times W\times C}$, and process these frames through the vision transformer (ViT) independently:

$$\{v_i^{cls}, v_i^1, v_i^2, ..., v_i^{patch}\} = ViT(\widetilde{v}_i), i = 1, 2, ..., N, \tag{1}$$

where *patch* represents the number of patches of the ViT. Utilizing the global feature v_i^{cls} as the feature for the i-th frame. The video LLM then applies a projection layer to map the ViT features into the feature space of the LLM:

$$\begin{aligned} z_i &= f(v_i^{cls}), i = 1, 2, ..., N, \\ Z &= \{z_i\} \in R^{N\times L\times d}, \end{aligned} \tag{2}$$

where Z is the input sequence LLM able to understand, d is the dimension of LLM's hidden space, and $N \times L$ is the total number of visual tokens for the input video. Noted that for the video LLM with different settings, the parameters N (*number of sampled frames*) and L (*number of tokens per frame*) from the above equations may

vary[1]. Since we require the coarse-grained video LLM to have the global understanding capability and temporal boundary perception capability on the input long videos. To achieve that, compared to the default setting of the original video LLM, we increase the *number of sampled frames* N to process more more video frames and gain a more comprehensive global understanding of the input long video while decreasing the *number of tokens per frame* L at a lower level, to ensure that $N \times L$ does not exceed the LLM processing limit for visual tokens.

After the processing of visual information, we introduce a prompt guidance P_c for **coarse-grained grounding** in the textual format, which is designed to stimulate the video LLM to temporally localize the most relevant short video segments to the textual question input and send the temporal grounding results. To enable the LLM aware of the original length of the input video before grounding prediction, we insert the statement *'The length of the original video is t_{total} seconds'* in the prompt P_c, where t_{total} is the total duration of the original long video. The prompt P_c and the original video question q are first combined and transformed into the textual embedding list $[w_1, w_2, ..., w_M] = Tokenizer(P_c, q)$, where $w_i \in R^d$ is the embedding of the word token and M is the word number. Then, the video feature sequence will be inserted to form the input content for the video LLM:

$$input = [Z, w_1, w_2, ..., w_M], \tag{3}$$

so that the video LLM can further encode the input embedding to understand the video, related question, and our coarse-grained prompt, and give the final response:

$$\begin{aligned} p_{grounding} &= VidLLM_{corase-grained}(input), \\ p_{grounding} &= \{t_{start}, t_{end}, answer\}, \end{aligned} \tag{4}$$

where $p_{grounding}$ is the output policy prompted in the JSON format and contains the predicted start timestamp t_{start} and end timestamp t_{end} of the video segment that the video LLM judges is most relevant to the original video question. Since video LLM with the coarse-grained module may encounter relatively simple video questions for short videos, it is also prompted allowed to give policies with direct output $answer$ in simple video cases.

Fine-grained Answering. With the predicted start timestamp t_{start} and end timestamp t_{end}, we obtain the video segment $v' \in R^{T' \times H \times W \times C}$ with $T' = T\frac{t_{end}-t_{start}}{t_{total}}$ frames. The fine-grained video LLM would sample N' frames represented as $\widetilde{v} \in R^{N' \times H \times W \times C}$ and process visual frames similar to the coarse-grained module:

$$\begin{aligned} \{v_i^{cls}, v_i^1, v_i^2, ..., v_i^{patch}\} &= ViT(\widetilde{v}_i), i = 1, 2, ..., N', \\ z_i &= f'(v_i^{cls}), i = 1, 2, ..., N', \\ Z &= \{z_i\} \in R^{N' \times L' \times d}. \end{aligned} \tag{5}$$

Since the fine-grained video understanding module is designed for video LLM to analyze the localized short video content related to the input query, which is relatively shorter, and its visual information is more worthy of in-depth extraction and understanding. Therefore, we increase the *number of tokens per frame* L compared to the default setting of the original video LLM, enabling the model to have fine-grained perception under visual frames from the relatively short video while decreasing the *number of sampled frames* N to keep $N \times L$ within the capabilities of LLM.

After the processing of visual information, we transform the original video question q into a textual embedding list $[w_1, w_2, ..., w_M] = Tokenizer(q)$ to encode the input embedding for the short video segment understanding:

$$\begin{aligned} input &= [Z, w_1, w_2, ..., w_M], \\ output &= VidLLM_{fine-grained}(input), \end{aligned} \tag{6}$$

and we obtain the detailed perception answer $output$ responded by the video LLM adapted with the fine-grained module, which analyzes the key video segment with more per-frame visual tokens.

3.2 Fine-to-Coarse Reflection

The fine-to-coarse reflection includes a fine-grained reflection model and a coarse-grained re-grounding model, which are consistent with the fine-grained video LLM and coarse-grained video LLM of the last section. Considering it is possible for the coarse-grained video LLM to produce incorrect grounding results in a single inference, during our fine-to-coarse reflection strategy, in addition to generating normal responses under fine-grained visual representations, the video LLM with the fine-grained module is also prompted to provide reflective judgments on the validity of the received short video segments and decide whether it requires re-grounding the relevant video segments.

Fine-grained Reflection. The visual information processing during the fine-grained reflection is the same as that of the fine-grained answering module. Different from the textual processing of fine-grained answering, we design prompt guidance P_f for **fine-to-coarse reflection prompt** in the textual format, which instructs the video LLM to provide reflective judgments on the validity of the received grounded video segments. The fine-grained prompt P_f and the original video question q would be combined and transformed into the textual embedding list $[w_1, w_2, ..., w_M] = Tokenizer(P_c, q)$ for short video understanding and reflection:

$$\begin{aligned} input &= [Z, w_1, w_2, ..., w_M], \\ p_{reflection} &= VidLLM_{fine-grained}(input), \\ p_{reflection} &= \{reflection, answer\}, \end{aligned} \tag{7}$$

where the output policy in the JSON format contains two aspects, which are $reflection$ and $answer$. The $reflection$ denotes the textual reason that the grounded short video is not suitable for the original question and requires another attempt for related segment grounding, and it would be returned to the coarse-grained re-grounding module. If the video LLM judges the localized video to be suitable for question answering, the $reflection$ would be empty and $answer$ would be the final output to the video question.

Coarse-grained Re-grounding. If the judgement of the fine-to-coarse reflection indicates that the grounded short video is not related to the video question, this $reflection$ information of the temporalization would be transformed into the format of *'t_{start} to t_{end} is not the suitable grounding segments because...'* and it will be returned to the coarse-grained video LLM.

The visual information processing during the coarse-grained re-grounding is the same as that of the coarse-grained grounding

[1]Normally LLMs have token limits. Therefore, the value of $N \times L$ cannot exceed a certain threshold.

module. During the textual processing, we will first combine the reflection and coarse-grained prompt to form a new coarse-grained grounding prompt $P_c = (P_c, reflection)$ with the last time grounding feedback. And the updated coarse-grained grounding prompt P_c is transformed similarly $[Z, w_1, w_2, ..., w_M] = Tokenizer(P_c, q)$ compared to the coarse-grained grounding module for re-grounding:

$$\begin{aligned} input &= [Z, w_1, w_2, ..., w_M], \\ p_{grounding} &= VidLLM_{coarse-grained}(input), \\ p_{grounding} &= \{t'_{start}, t'_{end}, answer\}, \end{aligned} \tag{8}$$

where the output policy $p_{grounding}$ contains the new start timestamps and end timestamps that the video LLM predicts are relevant to the original video question and different from the last time of coarse-grained grounding.

For a single video question in our complete ReCrossVLLM framework, the fine-grained reflection module would replace the fine-grained answering module, and the coarse-grained re-grounding module could be considered as a coarse-grained grounding module with at least one feedback prior. Therefore, two cross-granularity modules could be called multiple times, until the video LLM with the fine-grained module responds with positive feedback on the short video segment, or the video LLM with the coarse-grained module only responds with direct output policy without temporal grounding result. If the calls of two modules exceed a certain threshold limit, the calls will also terminate and select the answer response from the last round of video LLM as output.

4 Module Training

In this section, we will introduce how we optimize the modules of video LLM with different granularities. Our training method includes the fine-tuning of the coarse-grained module, the fine-tuning of the fine-grained module, and most importantly, the cross-granularity preference optimization for modules with different granularities through the coarse-to-fine strategy.

4.1 Coarse-grained and Fine-grained Fine-tuning

Formally, given a question q through tokenization and related video v through visual processing for video LLM, the supervised fine-tuning loss can be defined as the cross-entropy loss as follows:

$$L_{CE}(\widehat{y}(q,v), y) = -\sum_{(v,q,y)\in D} y \log(\widehat{y}(q,v)), \tag{9}$$

where y is the ground truth answer in the token sequence and D is the dataset.

Coarse-grained Module Training. To effectively localize short segments from the original video that are relevant to the question, we collect 175k training datasets of temporal video grounding datasets including Charades-STA [12], ActivityNet-Captions [17], and VTG-IT [14] to finetune the coarse-grained module, which will receive the global visual representations transformed from the complete video content and predict the most relevant short video segments to the question input.

Fine-grained Module Training. The fine-grained module is designed for video LLM to analyze the localized short video content related to the input query, which is relatively shorter, and its visual information is more worthy of in-depth extraction and understanding. Therefore, we select the training subset of 21.5k video data with a duration of less than one minute from several VideoQA datasets, including ActivityNet-QA [46], Ego-QA [13], and Next-QA [44], to fine-tune another fine-grained module to complete the fine-grained understanding and reasoning tasks for short videos.

4.2 Cross-granularity Preference Optimization

Since the task of providing temporal localization policies for video-related questions still has some differences from the retrieval input textual query, which the coarse-grained video LLM is trained with, it is difficult to use in-context learning alone to ensure the quality of policies. To address this issue, we propose a reinforcement learning-based cross-granularity training method to optimize cross-granularity modules of video LLM continuously for processing long videos, which enables the video LLM to improve the temporal grounding quality of generated policies, and adapt to the key video segment for fine-grained understanding.

Training Strategy. Given a question q about the input long video v, we prompt the coarse-grained video LLM to generate several different temporal localization policies $\{p_j\}$ of question-relevant short video segments from the original long video. These policies would inform the fine-grained module to process the grounded short videos in more detailed visual representations and give the final answers. It could be inferred that a correct policy that more accurately localizes the temporal video segments related to the video question would be more helpful for the fine-grained video LLM. Consequently, the loss computed by the video LLM tends to be smaller. Meanwhile, negative policies that localize irrelevant video segments would mislead the video LLM, resulting in incorrect responses and an increase in loss.

Therefore, we are able to provide feedback to the coarse-grained video LLM with the losses computed by the fine-grained video LLM for cross-granularity training. Inspired by the Reinforcement Learning from Human Feedback (RLHF) [3, 8], we apply Direct Preference Optimization (DPO) [28] to train the coarse-grained video LLM to generate more accurate policies. The DPO method directly optimizes the Video Large Language Model without explicit rewarding models and formulates the policy objective as:

$$\begin{aligned} &L_{DPO}(\pi_\theta; \pi_{ref}) \\ &= -E_{(q,v,p_w,p_l)\sim D}[\log \sigma(\beta \log \frac{\pi_\theta(p_w|q)}{\pi_{ref}(p_w|q)} - \beta \log \frac{\pi_\theta(p_l|q)}{\pi_{ref}(p_l|q)})], \end{aligned} \tag{10}$$

where p_w represents the positive policy with the smaller loss that accurately localizes the short video segment related to the video question, and p_l denotes the negative policy with the larger loss. π_θ represents our coarse-grained video LLM to be trained in this stage, while π_{ref} is a reference model also initialized with coarse-grained video LLM but remains frozen. σ is the sigmoid function and β is a controlling parameter.

The details about our reinforcement learning strategy are provided in Algorithm 1. Our cross-granularity preference optimization alternates between direct preference optimization for the coarse-grained video LLM and SFT for the fine-grained video LLM, which optimizes with the loss function in Equation 9 and 10. After the cross-granularity training, the coarse-grained video LLM is able to

Algorithm 1: Cross-granularity Training Strategy

Input: Coarse-grained module (coarse-grained video LLM) C, Temporal-Grounding module G, Coarse-grained prompt P_c, Fine-grained module (fine-grained video LLM) F, dataset $D = \{(v_i, q_i, y_i)\}_{i=1}^{N}$, training steps S, gradient accumulate step s, numbers of policies per data n

Output: Trained modules C and F

Freeze: F, G, **Activate:** C;
for $t = 1$ **to** S **do**
  initialize $\pi_\theta = C, \pi_{ref} = C$;
  for $m = 1$ **to** s **do**
    $i \leftarrow ((t-1)s + m - 1)\%N + 1$;
    Prepare data (v_i, q_i, y_i) from D;
    Generate policies $p_1, p_2, ...p_n = \pi_{ref}(P_c(q_i, v_i))$;
    for $j = 1$ **to** n **do**
      G execution localization: $v_j = G(v_i, p_j)$;
      F forward propagation: $\widehat{y} = F(q_i, v_j)$;
      Compute L_{CEj} in Equation (9);
    end
    $p_w \leftarrow \arg\min_{\{p_j\}} L_{CE}$;
    $p_l \leftarrow \{p_j, p \neq p_w\}$;
    Optimize π_θ with loss: L_{DPO} in Equation (10);
    Add (v_i, q_i, y_i) to $CACHE$;
  end
  $C \leftarrow \pi_\theta$, freeze C, activate F;
  for $i = 1$ **to** s **do**
    Prepare data (v_i, q_i, y_i) from $CACHE$;
    Generate policies single policy $p = Q(P_c(q_i, v_i))$;
    G execution localization: $v_p = G(v_i, p)$;
    F forward propagation: $\widehat{y} = F(q_i, v_p)$;
    Optimize F with loss: L_{CE} in Equation (9);
  end
  clear $CACHE$, freeze F, activate C;
end

provide refined policies with more accurate localization segments, and the fine-grained video LLM would also adapt the localization of short videos from the coarse-grained video LLM. For the reinforcement learning, we select a 7.5k training dataset from the NeXT-QA and ActiveNet-QA datasets with video data exceeding 2 minutes.

5 Experiments

Implementations. We implement our fine-tuning and preference optimization method based on the SWIFT framework. We utilize LLaVA-Next-Video(7B) [50] and LLaVA-Video(7B) [51] as different backbone settings. For LLaVA-Next-Video(7B), we set the *number of sampled frames* N and *number of tokens per frame* L to $(16, 12\times12 = 144)$ for fine-grained video LLM and $(144, 4 \times 4 = 16)$ for coarse-grained video LLM during training, and extended to $(32, 12 \times 12 = 144)$ and $(288, 4 \times 4 = 16)$ during inference enabled by linear scale factor. For LLaVA-Video(7B), we set $(32, 13 \times 13 = 169)$ for the fine-grained video LLM and $(256, 4 \times 4 = 16)$ for coarse-grained video LLM through training and inference. Before the fine-tuning of different granularity modules and reinforcement learning, we first train the visual adapter f through image-text feature alignment. The overall module training is conducted under 8 NVIDIA A100-40GB GPUs.

Evaluation. We conduct our experiments on long video understanding and temporal video grounding datasets. We select VideoMME, Lvbench, and MLVU for assessing the long video understanding ability [10, 37, 52]. For the temporal video grounding task, we utilize the validation set of ActivityNet Captions(val_2) and the test set of Charades-STA. Our baselines include long-video video LLMs and fine-grained video LLMs for long video understanding, and baselines for temporal video grounding include state-of-the-art temporal perception video LLMs.

Baselines. On long video understanding tasks, our baselines include long-video video LLMs LongVA, LongVU, Video-RAG, video-XL, video-XL2 [24, 26, 31, 32, 49] and fine-grained video LLMs such as Video-LLaVA, VideoChat2, Chat-UniVi-V1.5, ShareGPT4Video, LLaVA-NeXT-Video, LLaVA-Video and Qwen2.5-VL [2, 6, 16, 19, 21, 50, 51]. Our baselines for temporal video grounding include temporal perception video LLMs such as VTimeLLM [15], TimeChat [29], Monmentor [25], VTG-LLM [14], NumPro [43], ChatVTG [27], and the LLM-based temporal grounding method BTDP [9].

5.1 Experiments on Long Video Understanding

Based on the long video understanding benchmarks, we evaluate the capabilities of existing video LLMs in long video understanding tasks. As the results shown in Table 1, we draw the following conclusions. First, our ReCrossVLLM consistently achieves the best overall performance across all benchmarks: with the LLaVA-Video-7B backbone, it attains 71.9% on VideoMME (Overall), surpassing Qwen2.5-VL (71.6%) and LLaVA-Video (69.7%), and also sets new state-of-the-art results on Lvbench (48.9%) and MLVU (76.1%). Notably, even using the smaller LLaVA-NeXT-Video-7B backbone (65.7%), our method outperforms the larger LLaVA-NeXT-Video-34B (54.9%), demonstrating the effectiveness of our cross-granularity design.

Second, on VideoMME, as video duration increases from short to long, all methods exhibit performance drops, yet ReCrossVLLM shows the smallest relative decline (79.8% → 64.7%) compared to Qwen2.5-VL (81.4% → 62.6%) and LLaVA-Video (78.0% → 61.8%), thanks to our coarse-to-fine grounding and fine-to-coarse reflection strategies. Third, beyond VideoMME, ReCrossVLLM excels on Lvbench (extreme long videos) and MLVU, outperforming strong baselines such as Video-XL2 and LongVU, confirming its generalization and robustness across diverse video lengths and task formats.

5.2 Experiments on Temporal Video Grounding

We evaluate the capabilities of existing video LLMs in temporal video grounding tasks, with results shown in Table 2. Given a video and a textual query, the models must predict the start and end timestamps of the corresponding segment. In our cross-granularity inference, the coarse-to-fine strategy first localizes a short candidate segment. The fine-grained video LLM then receives a description that this short video is sampled from a specific time interval of the original video, enabling it to refine the localization under more detailed visual representations. As seen in the table, our ReCrossVLLM consistently outperforms all prior video LLMs trained for temporal

Table 1: Performance comparison of state-of-the-art video LLMs with our ReCrossVLLM methods on long video understanding benchmarks. The best average performance is in bold and the second is underlined.

Models	Size	VideoMME				Lvbench	MLVU
		Short	Medium	Long	Overall		
Duration(min)		≤2	4~15	30~60	1~60	30~140	3~120
Video-LLaVA [21]	7B	46.1	40.7	38.1	41.6	21.6	47.3
Chat-UniVi [16]	7B	51.2	44.6	41.8	45.9	25.3	52.6
ShareGPT4Video [6]	8B	53.6	39.3	37.9	43.6	21.8	46.4
VideoChat2 [19]	7B	52.8	39.4	39.2	43.8	23.7	47.9
LongVA [49]	7B	61.6	53.6	47.6	54.3	31.7	56.3
Video-RAG with LLaVA-NeXT [24]	7B	56.6	47.4	46.0	50.0	30.2	53.5
LongVU [31]	7B	64.7	58.2	59.5	60.9	38.3	65.4
Video-XL [32]	7B	67.4	60.7	54.9	61.0	37.7	64.9
Video-XL2 [26]	8B	73.7	65.9	60.2	66.6	<u>48.4</u>	74.8
VITA 1.5 [11]	7B	67.0	54.2	47.1	56.1	32.1	60.2
LLaVA-NeXT-Video [50]	34B	65.1	52.2	47.2	54.9	32.2	61.6
LLaVA-Video [51]	7B	78.0	69.3	61.8	69.7	43.2	70.8
Qwen2.5-VL [2]	7B	**81.4**	<u>70.8</u>	<u>62.6</u>	<u>71.6</u>	45.3	<u>75.5</u>
Ours-CrossVLLM w/ LLaVA-Next-Video	7B	70.9	64.4	61.9	65.7	41.8	70.4
Ours-CrossVLLM w/ LLaVA-Video	7B	<u>79.8</u>	**71.2**	**64.7**	**71.9**	**48.9**	**76.1**

Table 2: Performance comparison of state-of-the-art temporal perception video LLMs with our ReCrossVLLM methods on temporal video grounding tasks. The best average performance is in bold and the second is underlined.

Models	ActivityNet Captions				Charades-STA			
	R@0.3	R@0.5	R@0.7	mIoU	R@0.3	R@0.5	R@0.7	mIoU
ChatVTG [27]	40.7	22.5	9.4	27.2	52.6	33.0	15.9	34.9
VtimeLLM [15]	44.0	27.8	14.3	30.4	51.0	27.5	11.4	31.2
TimeChat [29]	-	-	-	-	-	32.2	13.4	-
Momentor [25]	42.9	23.0	12.4	29.3	42.6	26.6	11.6	28.5
VTG-LLM [14]	-	-	-	-	52.0	33.8	15.7	-
NumPro [43]	45.5	<u>30.8</u>	<u>18.4</u>	33.6	<u>60.7</u>	36.8	15.9	38.5
BTDP [9]	<u>50.6</u>	30.6	17.5	<u>36.6</u>	58.3	<u>40.0</u>	<u>20.9</u>	<u>39.1</u>
Ours-ReCrossVLLM w/ LLaVA-Next-Video	**53.9**	**41.7**	**23.1**	**39.8**	**63.5**	**44.9**	**21.0**	**42.6**

perception on both ActivityNet Captions and Charades-STA across all recall thresholds and mean IoU.

Specifically, on ActivityNet Captions, ReCrossVLLM achieves 53.9% (R@0.3), 41.7% (R@0.5), and 23.1% (R@0.7), substantially exceeding the previous best method BTDP. On Charades-STA, our method attains 63.5% (R@0.3), 44.9% (R@0.5), and 21.0% (R@0.7), demonstrating strong boundary localization capability. The fine-to-coarse reflection effectively mitigates error propagation from initial coarse localization, while fine-grained tokens preserve critical details for accurate boundary detection. These results confirm that our cross-granularity design generalizes robustly from long video understanding to temporal grounding. Notably, the consistent gains across both datasets also demonstrate the generalization abilities of our approach for fine-grained temporal reasoning tasks.

5.3 Ablation Study

In this section, we provide detailed ablation analyses of our cross-granularity strategies through experiments on our models to evaluate the effectiveness of our designed modules. The results are shown in Table 3.

5.3.1 Result about Fine-tuning. Based on the results of Row 1,3,4 and 5 from Table 3, we can see the positive impact through both coarse-grained and fine-grained fine-tuning on video LLMs. The utilization of coarse-grained fine-tuning comprehensively improves the video LLMs in processing both temporal video grounding and long video understanding tasks. The fine-grained fine-tuning helps mainly in short video understanding and temporal video grounding. In addition, although only applying basic model fine-tuning does not significantly improve overall performance compared to applying our entire cross-granularity framework, they still occupy an

Table 3: Ablation study of our reflective cross-granularity strategies with LLaVA-Next-Video. The coarse-to-fine grounding strategy is activated through the entire experiment. The *Tune* in the line of *Coarse* and *Fine* respectively represent coarse-grained training and fine-grained training for video LLMs while *Freeze* represents image-text feature alignment only. *RL* represents the utilization of cross-granularity preference optimization. The best average performance is in bold.

Row	Coarse	Fine	RL	Fine-to-Coarse Reflection	ActivityNet Captions				VideoMME			
					R@0.3	R@0.5	R@0.7	mIoU	Short	Medium	Long	Overall
1	Freeze	Freeze	✗	✗	31.3	16.0	6.8	22.4	55.2	44.0	42.7	47.3
2	Freeze	Freeze	✓	✓	31.8	16.6	6.9	23.2	56.8	44.6	41.5	47.6
3	Tune	Freeze	✗	✗	42.3	25.8	11.6	29.4	58.2	45.4	43.7	49.1
4	Freeze	Tune	✗	✗	38.7	20.4	8.6	25.9	60.3	44.5	42.2	49.0
5	Tune	Tune	✗	✗	44.7	28.1	14.2	31.2	61.6	47.2	44.9	51.2
6	Tune	Tune	✓	✗	52.3	38.9	20.4	38.5	**71.8**	60.3	53.8	62.0
7	Tune	Tune	✗	✓	47.9	30.5	15.6	32.4	67.7	57.8	51.4	58.9
8	Tune	Tune	✓	✓	**53.9**	**41.7**	**23.1**	**39.8**	70.9	**64.4**	**61.9**	**65.7**

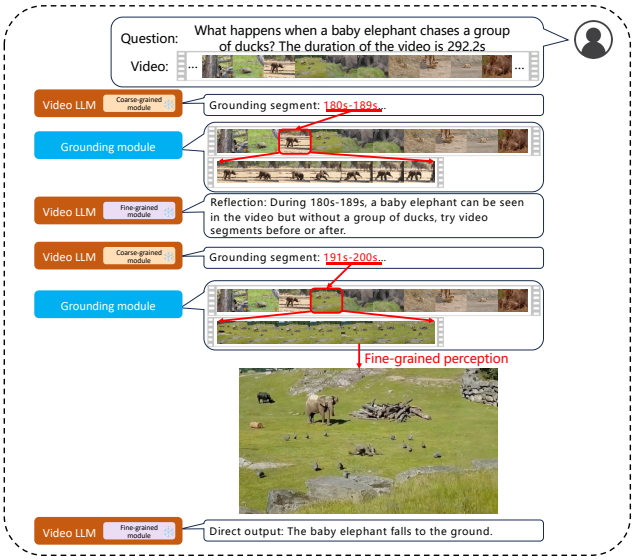


Figure 3: Example with fine-to-coarse reflection. For the shown long video understanding task on VideoMME, our reflection strategy helps in re-grounding the correct video segment after grounding the incorrect segment for the first time in the coarse-to-fine strategy.

important position in our method, as shown in Row 1 and 2, we can see that directly using not fine-tuned video LLMs for fine-to-coarse reflection and reinforcement learning through cross-granularity preference optimization would not bring performance improvement, encountering a certain degree of cold start problem.

5.3.2 Result about Cross-granularity Preference optimization. In Row 5, 7 vs Row 6, 8 from Table 3, we removed the cross-granularity preference optimization strategy and conducted evaluations on benchmarks. The results show a decrease in all the evaluation metrics. Compared to the results of other models shown in Table 1 and 2, our experiment without the preference optimization method only performs at an average level among baselines, indicating the necessity of the cross-granularity preference optimization strategy we proposed for training video LLMs to further improve grounding efficiency.

5.3.3 Result about Fine-to-Coarse Reflection. During the inference stage, our fine-to-coarse reflection strategy prompts the fine-grained video LLM to reflect on the locating effectiveness and decide whether to return the response to the coarse-grained video LLM for re-grounding with reflection feedback. As shown in Row 7, 8 vs Row 5, 6, with the addition of fine-to-coarse reflection, the performance of our method has further improved, indicating our fine-to-coarse reflection provided by fine-grained video LLM contains useful information for coarse-grained video LLM to regenerate a suitable grounding policy. Analyses of a detailed example shown in Figure 3 also support the positive effects of our fine-to-coarse reflection strategy. In the example, the grounding condition is *'a baby elephant chases a group of ducks'* and the video includes several segments distributed in different times but all contain *'elephant'*, which interferes with the Video LLM to localize the correct video segment at the first time, while our designed fine-to-coarse reflection helps to re-localize the correct key segment.

6 Conclusion

In this paper, we propose ReCrossVLLM, a novel reflective cross-granularity video LLM framework for long video understanding. Specifically, we design the coarse-to-fine grounding and fine-to-coarse reflection strategies utilizing adaptive modules of video LLM with different granularities to collaborate with each other for cross-granularity long video inference. We further propose a cross-granularity preference optimization strategy to optimize video LLM through training and inference when processing long videos. Extensive experiments demonstrate that ReCrossVLLM outperforms existing video LLMs in long video question answering and temporal video grounding tasks, indicating its superiority for processing both high-level semantics and low-level visual details in long video understanding.

Acknowledgments

This work is supported by National Natural Science Foundation of China (No.62222209), and Beijing National Research Center for Information Science and Technology under Grant No.BNR2026TD03005.

References

[1] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966* (2023).

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923* (2025).

[3] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073* (2022).

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

[5] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. 2023. Fuyu-8B: A multimodal architecture for AI agents.

[6] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, et al. 2024. Sharegpt4video: Improving video understanding and generation with better captions. *arXiv preprint arXiv:2406.04325* (2024).

[7] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. 2024. VideoLLaMA 2: Advancing Spatial-Temporal Modeling and Audio Understanding in Video-LLMs. *arXiv preprint arXiv:2406.07476* (2024).

[8] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[9] Xiongwen Deng, Haoyu Tang, Han Jiang, Qinghai Zheng, and Jihua Zhu. 2025. Boundary-Aware Temporal Dynamic Pseudo-Supervision Pairs Generation for Zero-Shot Natural Language Video Localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 2717–2725.

[10] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. 2024. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075* (2024).

[11] Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, et al. 2025. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *arXiv preprint arXiv:2501.01957* (2025).

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

[13] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. 2022. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18995–19012.

[14] Yongxin Guo, Jingyu Liu, Mingda Li, Dingxin Cheng, Xiaoying Tang, Dianbo Sui, Qingbin Liu, Xi Chen, and Kevin Zhao. 2025. Vtg-llm: Integrating timestamp knowledge into video llms for enhanced video temporal grounding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 3302–3310.

[15] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. 2024. Vtimellm: Empower llm to grasp video moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14271–14280.

[16] Peng Jin, Ryuichi Takanobu, Wancai Zhang, Xiaochun Cao, and Li Yuan. 2024. Chat-univi: Unified visual representation empowers large language models with image and video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13700–13710.

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

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

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

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

[21] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2024. Video-LLaVA: Learning United Visual Representation by Alignment Before Projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. 5971–5984.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 740–755.

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in neural information processing systems* 36 (2024).

[24] Yongdong Luo, Xiawu Zheng, Xiao Yang, Guilin Li, Haojia Lin, Jinfa Huang, Jiayi Ji, Fei Chao, Jiebo Luo, and Rongrong Ji. 2024. Video-rag: Visually-aligned retrieval-augmented long video comprehension. *arXiv preprint arXiv:2411.13093* (2024).

[25] Long Qian, Juncheng Li, Yu Wu, Yaobo Ye, Hao Fei, Tat-Seng Chua, Yueting Zhuang, and Siliang Tang. 2024. Momentor: Advancing Video Large Language Model with Fine-Grained Temporal Reasoning. In *Forty-first International Conference on Machine Learning*.

[26] Minghao Qin, Xiangrui Liu, Zhengyang Liang, Yan Shu, Huaying Yuan, Juenjie Zhou, Shitao Xiao, Bo Zhao, and Zheng Liu. 2025. Video-XL-2: Towards Very Long-Video Understanding Through Task-Aware KV Sparsification. *arXiv preprint arXiv:2506.19225* (2025).

[27] Mengxue Qu, Xiaodong Chen, Wu Liu, Alicia Li, and Yao Zhao. 2024. ChatVTG: Video Temporal Grounding via Chat with Video Dialogue Large Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1847–1856.

[28] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2024).

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

[30] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems* 35 (2022), 25278–25294.

[31] Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, et al. 2025. LongVU: Spatiotemporal Adaptive Compression for Long Video-Language Understanding. In *Forty-second International Conference on Machine Learning*.

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

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

[34] Xichen Tan, Yuanjing Luo, Yunfan Ye, Fang Liu, and Zhiping Cai. 2025. ALLVB: All-in-One Long Video Understanding Benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 7211–7219.

[35] Xi Tang, Jihao Qiu, Lingxi Xie, Yunjie Tian, Jianbin Jiao, and Qixiang Ye. 2025. Adaptive keyframe sampling for long video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 29118–29128.

[36] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).

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

[38] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. 2024. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*. Springer, 58–76.

[39] Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. 2025. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 3272–3283.

[40] A Waswani, N Shazeer, N Parmar, J Uszkoreit, L Jones, A Gomez, L Kaiser, and I Polosukhin. 2017. Attention is all you need. In *NIPS*.

[41] Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. 2024. Longvlm: Efficient long video understanding via large language models. In *European Conference on Computer Vision*. Springer, 453–470.

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

[43] Yongliang Wu, Xinting Hu, Yuyang Sun, Yizhou Zhou, Wenbo Zhu, Fengyun Rao, Bernt Schiele, and Xu Yang. 2025. Number it: Temporal grounding videos like flipping manga. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 13754–13765.

[44] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. 2021. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9777–9786.

[45] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. 2023. Self-chained image-language model for video localization and question answering. *Advances in Neural Information Processing Systems* 36 (2023), 76749–76771.

[46] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. 2019. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9127–9134.

[47] Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. 2024. A simple llm framework for long-range video question-answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. 21715–21737.

[48] Hang Zhang, Xin Li, and Lidong Bing. 2023. 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*. 543–553.

[49] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. 2024. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852* (2024).

[50] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. 2024. LLaVA-NeXT: A Strong Zero-shot Video Understanding Model. https://llava-vl.github.io/blog/2024-04-30-llava-next-video/

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

[52] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, et al. 2025. Mlvu: Benchmarking multi-task long video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*. 13691–13701.