

Flow4Agent: Long-form Video Understanding via Motion Prior from Optical Flow

Ruyang Liu Shangkun Sun Haoran Tang Wei Gao ✉ Ge Li ✉

School of Electronic and Computer Engineering, Shenzhen Graduate School, Peking University

{ruiyang@stu., sunshk@stu., hrtang@stu., gaowei262@, geli@}pku.edu.cn

Abstract

*Long-form video understanding has always been a challenging problem due to the significant redundancy in both temporal and spatial contents. This challenge is further exacerbated by the limited context length of Multimodal Large Language Models (MLLMs). To address this issue, many previous works have attempted to extract key video information, where the “key” is typically semantic-aware and heavily dependent on the CLIP model as prior. In this paper, we propose **Flow4Agent**, a novel framework that pioneeringly incorporates motion priors from optical flow to facilitate LLM-based long video understanding. Flow4Agent mitigates the redundancy in long videos at both temporal and spatial levels through two core modules: **Temporal Granularity Optimization (TGO)** adaptively refines frame-level hierarchies, which first leverages coarse flow priors to group similar visual contents and then applies semantic priors to filter out highly irrelevant scene information. **Motion Token Pruning (MTP)** further refines the intra-frame visual representations, pruning high-redundancy video tokens using fine-grained optical flow information. Extensive experiments demonstrate that our Flow4Agent outperforms existing methods across a wide range of video MLLM benchmarks, especially for hour-level video understanding tasks, achieving 64.7% on Video-MME, 71.4% on MLVU and 60.4% on LongVideoBench.*

1. Introduction

Multimodal Large Language Models (MLLMs) have made significant strides recently. Thanks to the advancements in LLM [9, 14, 50] and multimodal [5, 16, 40] pretraining, current MLLMs can effectively interpret visual sequences in images and videos. These models typically support video sequences with hundreds of frames, which is sufficient to cover all contents in short videos, using uniform sampling with whether fixed frame numbers or fixed fps. However, for hour-long videos, this implies that at least one-minute

video is distributed to only one frame, leading to substantial information loss, as shown in the first line of Fig. 1.

To enable MLLMs to process more video content, one approach is to resample and compress the video tokens [15, 21, 27, 48]. However, dense resampling inevitably causes the loss of visual information, while frames that can be accommodated by the MLLM remain constrained by a clear upper limit. Another approach is to use memory structures [38, 53] or context extension [24, 54], enabling the LLM to process densely sampled video frames. However, this method overlooks the widespread information redundancy in long videos. As shown in the second line of Fig. 1, the significant redundancy in both time (irrelevant video frames) and space (repetitive content within the same scene) can overwhelm the LLM, resulting in mistakes during long video understanding.

To address the ubiquitous redundancy in videos, an intuitive solution is to extract key video information. This typically requires additional priors, with the most common being semantic information, such as using the CLIP model to retrieve relevant video content [12, 32, 43, 48] or feeding the video’s dense captions into the LLM for further reasoning and judgment [42, 43, 45]. However, this reliance on semantic priors has two major drawbacks. First, it heavily depends on the information provided in the user’s instructions; when the query offers limited details, much of the effectiveness is lost. Second, the method’s performance is constrained by the prior model, such as CLIP models or captioning models, meaning that errors in these models can significantly distort subsequent understanding.

In this paper, we introduce a previously overlooked prior, the motion information from optical flow, to assist in extracting key video content, and propose a novel method, Flow4Agent. Flow4Agent refines the key content in two aspects: inter-frame and intra-frame, which are addressed by the Temporal Granularity Optimization (TGO) module and the Motion Token Pruning (MTP) module, respectively. Specifically, the TGO module utilizes efficient coarse optical flow to accurately cluster video scenes, and on this basis, it leverages semantic priors to obtain a sufficiently dis-

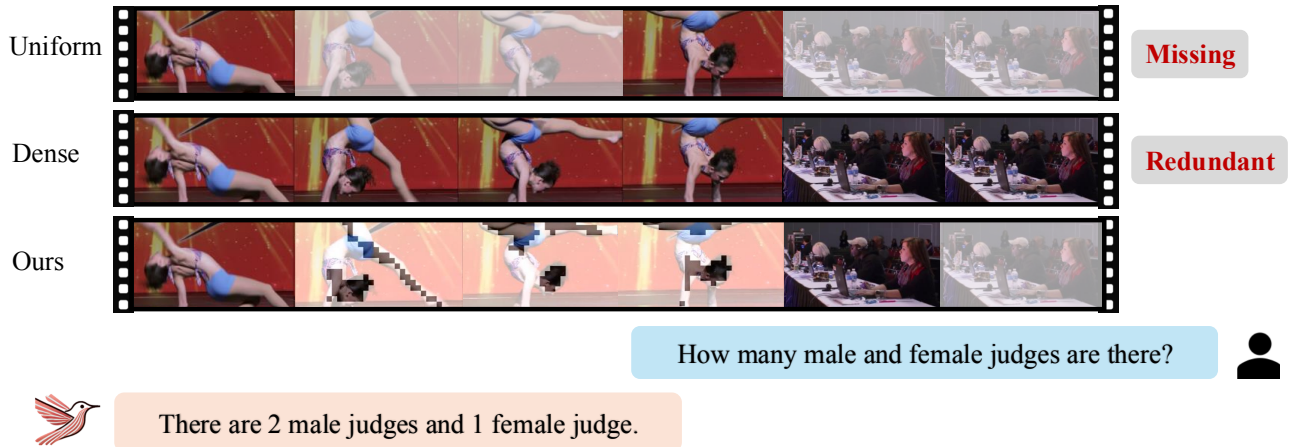


Figure 1. Comparison between uniform sampling, dense sampling, and our proposed Flow4Agent.

tinctive set of scenes, thus obtaining highly representative inter-frame features. Compared to methods that rely solely on semantic priors, our approach is more robust, as we do not depend on semantic priors to directly obtain keyframes. Instead, we use semantic priors to filter out entirely irrelevant scenes, resulting in a lower p-value. The MTP module, on the other hand, addresses intra-frame redundancy. For highly overlapping video features within the same scene, it uses fine-grained optical flow to filter out the more noteworthy and dynamically varying representations, thereby reducing the overall video redundancy. As the first model to incorporate optical flow for LLM-based video understanding, Flow4Agent neither requires dense captions nor depends on the user-provided details, enabling it to acquire more robust key video content at a lower cost.

To evaluate Flow4Agent, we conducted extensive experiments across a broad range of video understanding benchmarks, including VideoMME [10], EgoSchema [30], Perceptiontest [33], MLVU [57], NextQA [47], and LongVideoBench [46]. The experimental results demonstrate that Flow4Agent achieves state-of-the-art performance compared to other recent models. Notably, on the three benchmarks with significantly longer videos—VideoMME, MLVU, and LongVideoBench—Flow4Agent attained leading scores of 64.7%, 71.4%, and 60.4%, respectively. Additionally, we evaluated Flow4Agent on various foundational models, further confirming the effectiveness of our motion priors for long-form video comprehension.

The main contributions of our paper are summarized as:

- We propose Flow4Agent, which, to the best of our knowledge, is the first model to utilize optical flow information for LLM-based video understanding.
- We present two novel modules, Temporal Granularity Optimization and Motion Token Pruning, which leverage optical flow from coarse to fine to extract key video content

both inter-frame and intra-frame.

- We conducted extensive experiments on a wide range of video understanding benchmarks, validating the superior performance of Flow4Agent in video understanding, particularly in long video comprehension.

2. Related Work

Video-based Large Language Models. Recent advances in multi-modal large language models (MLLMs) have shown significant progress in processing multi-modal inputs, including video data. Existing research on Video LLMs has focused on both data and model aspects. Regarding data, it has evolved from initially using only video instruction data [19, 28, 29] to incorporating mixed multi-modal data [16, 17, 41, 56, 58], which has proven highly effective for video understanding. Additionally, further Reinforcement Learning from Human Feedback (RLHF) after instruction tuning has also demonstrated significant effectiveness for video dialogue [1, 2, 55]. At the model level, improvements in the performance of Video LLMs largely depend on advancements in the pretraining of individual components. For example, the more advanced SigLiP [51] surpasses CLIP [35] as a visual encoder; models based on the Qwen series [50] achieve better results than those using Vicuna [7] as LLM; and Video LLMs built on the latest generation of LLaVA [16, 22, 23] consistently outperform earlier versions of LLaVA and the BLIP series [8, 18]. Meanwhile, compared to the initial mean pooling strategy [28, 29, 52] and later approaches incorporating various video-specific components [15, 20, 21, 25], a more mainstream method has emerged: simply resampling video tokens to directly form a spatiotemporal sequence as input to the LLM [16, 23, 26, 48, 56, 58]. While this paradigm has shown promising results on short videos, the lengthy video sequences and the limited context window of LLMs hinder effective comprehension of long videos.

Long Video Understanding. Fixed-frame sampling remains the predominant choice for most methods. Although some studies have demonstrated the effectiveness of fixed-FPS sampling [21, 36, 58], the limited context window of LLMs ultimately constrains the number of frames that can be processed. To address this issue, various approaches have been proposed. For example, MovieChat [38] and Flash-VStream [53] adopt memory structures and sliding windows to enable streaming video input, while LongVA [54] and Kangaroo [24] extend the LLM’s context length to accommodate more frames. A more common strategy is video frame resampling, where techniques such as Perceiver-style query sampling [21, 26, 58], clustering [15], and simple local average pooling [16, 27, 48, 49, 56] all effectively reduce the number of frames. However, long videos inherently contain significant redundancy—temporally, only a few frames carry meaningful information, while spatially, adjacent frames often exhibit high similarity. Thus, feeding the entire long-from video into the LLM is unnecessary.

To this end, a common approach is to extract key content from videos, where the definition of “key” is guided by additional prior information. For example, LVNet [32] and LongVU [36] utilize extra visual encoders to compute the similarity between frame features. A more prevalent strategy leverages semantic priors, typically derived from pretrained retrieval and captioning models [12, 27, 36, 42, 43, 45]. A representative example is VideoAgent [43], which employs both dense captions and CLIP to identify key frames relevant to the user’s query. However, as discussed in Section 1, methods relying on semantic priors highly depend on the informative user instruction and accurate prior models. In contrast, our proposed Flow4Agent introduces motion priors from optical flow for the first time, reducing excessive dependence on semantic priors. Additionally, motion information enables a more precise elimination of redundant content.

3. Approach

In this section, we will elaborate on how Flow4Agent leverages optical flow to drive LLM-based video understanding. Optical flow has long been an important video understanding prior that provides motion information. In the past, while some previous works have used optical flow for action recognition [37, 39] or dataset sample filtering [4], no research has directly utilized optical flow for LLM-based video understanding. As shown in Fig. 2, Flow4Agent addresses redundancy and extracts key content in both inter-frame and intra-frame aspects. In Section 3.1, the Temporal Granularity Optimization (TGO) module uses HSV transformation with coarse optical flow priors to cluster video content, and employs semantic priors for hypothesis testing to identify representative video content. Then, in Section

3.2, the Motion Token Pruning (MTP) module uses fine-grained optical flow to capture more significant motion features spatially within high-redundancy events.

3.1. Temporal Granularity Optimization

Uniform frame sampling, whether using fixed frame numbers or fixed FPS, has been a prevalent approach in previous video understanding models. However, this method often overlooks temporal structural dynamics, leading to limited content diversity. For instance, events with a larger number of frames are likely to be sampled more heavily, even if they lack significant dynamic changes, resulting in redundant frames that contribute little additional information. Conversely, events with fewer frames may be overlooked, even though they might contain crucial information. To address this issue, we propose Temporal Granularity Optimization (TGO) to adaptively refine temporal representation hierarchies in video analysis. As illustrated in Figure 2, the core of TGO lies in a dual-phase spatiotemporal decomposition.

Dynamic Event Split. We design a motion-aware chromatic analysis strategy to partition a video into different temporal units. The partitioning strategy consists of two stages. In the first stage, we utilize projected temporal differences to divide dynamic events in a coarse manner. Inspired by [11], motion variations of RGB space pixels are often susceptible to factors like illumination changes. Therefore, we first transform frames into the HSV color space, which is less sensitive to illumination variations, thereby eliminating the impact of motion-irrelevant factors like lighting on pixel values. In this space, changes in pixel values better reflect actual event dynamics. We then compute the mean square error between consecutive frames, and if it exceeds a threshold, we temporarily mark it as a boundary, completing the coarse first-stage screening. Given the input video V comprised of frames I_1, \dots, I_N , The first-stage process can be formulated as:

$$V' = \{\Phi(I_t) \mid t \in \mathbb{N}\}, \quad (1)$$

$$\Delta V' = \{\|I'_{t+1} - I'_t\|_2 \mid t \in \mathbb{N}\}, \quad (2)$$

$$C = \{I_t \mid \Delta V'_t > \theta, t \in \mathbb{N}\}, \quad (3)$$

where Φ refers to the HSV transformation and I' denotes the transformed frames. θ represents the threshold that filters static frames, and C is the set of coarsely selected boundaries in this stage. After that, we leverage pixel-level motion information provided by optical flow to achieve more precise partitioning of dynamic events. For each potential temporal boundary identified in the first stage, we calculate adjacent M flows within a temporal window. If the maximum magnitudes among these M optical flows exceed a specific threshold, we designate the corresponding frame in the window as the final temporal boundary. In practice, M is set to 3. We employ SeaRAFT [44] for flow

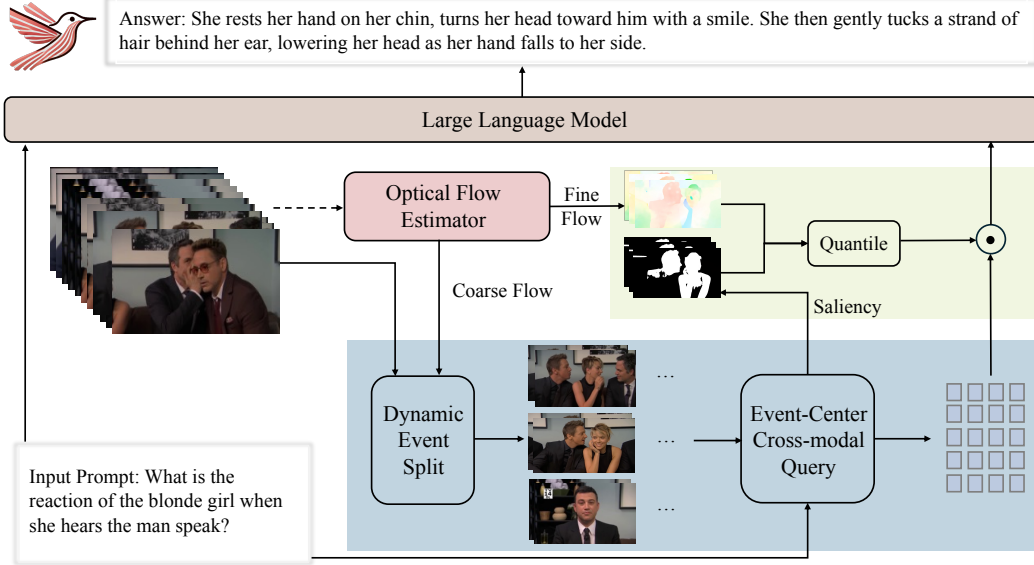


Figure 2. Overview of the proposed Flow4Agent. TGO and MTP strategies are highlighted in the blue and yellow regions, respectively. The dashed line indicates the frames after the first stage in DES, and \odot denotes the Hadamard product operator.

calculation, which enables efficient estimation through non-overlapping estimation. Notably, we iterate for only a few rounds to obtain a coarse optical flow, which is sufficient for precise boundary splitting. The entire process can be formulated as:

$$W = \left\{ \left\{ I_k \right\}_{k=t-\lfloor \frac{M}{2} \rfloor}^{t+\lfloor \frac{M}{2} \rfloor} \mid I_t \in C \right\}, \quad (4)$$

$$F = \{ \mathcal{E}(w) \mid w \in W \}, \quad (5)$$

$$S = \{ I_{\kappa(f_i)} \mid i = \arg \max_{1 \leq j \leq M} f_j > \eta, f \in F \}, \quad (6)$$

where I_t denotes the boundary frame in the first stage with the index of t in the original video. w represents a set of frames in the selected window, and \mathcal{E} refers to the function calculating the corresponding optical flows. f denotes a set of flows in the selected window, and η represents the threshold. κ is the look-up function that derives the index of the flow in the original video. Traditional key content extraction models often rely on frame-based semantic retrieval or dense captioning. In contrast, once we have divided the video into events, event-centered operations are more efficient and yield better results compared to frame-centered ones, as clustering frames inherently involves the removal of redundant information.

Event-Center Cross-modal Query. After dividing the video into events $\{S_i\}$, we select the middle frame of each event to form the anchor frames. This is because within the same event, semantic scenes largely remain unchanged, and thus, frames within the same event are almost identical for an image model. Based on this, we use semantic priors to select keyframes. Previous models typically rely directly on image retrieval models to obtain the most relevant results.

Specifically, the target events can be formulated as:

$$S_{out} = \{ S_j \mid j = \operatorname{argtopk}_{f_i \subseteq S_i} (\Theta_Q(f_i) \cdot \Theta_Q(q)) \}, \quad (7)$$

where f_i is the anchor frame of event S_i , q is the embedding of the user’s instruction, and Θ_Q represents the parameters of image retrieval models such as CLIP or SigLIP.

However, this approach is highly dependent on the accuracy of the prior model Θ_Q . While query-related events may exhibit high similarity, they do not necessarily have the top-k largest similarity scores. Therefore, we introduce two constraints for event selection: first, the selected events should be significant enough to represent the entire video; second, we aim to select as few events as possible to maximize redundancy removal. Therefore, the selected event set S_{out} needs to satisfy the following constraints:

$$\begin{cases} \min \operatorname{len}(S_{out}), \\ \alpha(S_i) = \frac{e^{\Theta_Q(f_i) \cdot \Theta_Q(q)}}{\sum_j e^{\Theta_Q(f_j) \cdot \Theta_Q(q)}}, \\ \text{p-value} = 1 - \sum_{S_i \subseteq S_{out}} \alpha(S_i) < 0.05, \end{cases} \quad (8)$$

where $\alpha(S_i)$ defines the significance level of the temporal event. When an event contains content strongly relevant to the user’s instruction, its significance will outweigh that of other events, and it will be selected independently. Conversely, when the user’s instruction lacks sufficient details, these constraints ensure that all important scenes are not overlooked, while filtering out scenes with excessively low significance that are completely irrelevant. Thanks to the integration of motion priors with semantic priors, we are able to adopt a more conservative strategy that filters out a significant amount of redundant content without missing important information.

3.2. Motion Token Pruning

Compared to the numerous works on frame selection strategies for inter-frame redundancy, methods addressing intra-frame redundancy are relatively rare. For example, [27] and [12] use language information to further retrieve key tokens within frames, while [36] employs DINOv2 [31] to filter out spatial tokens highly similar to the anchor frame. However, these models also overlook critical motion information. In the same scenes, most of the background remains unchanged, while the small amount of changing foreground information is key to understanding. Therefore, we propose the Motion Token Pruning (MTP) strategy for intra-frame sampling, which utilizes fine-grained motion information to further prune the content within frames.

Specifically, we find that optical flow naturally describes the dynamic information in the scene. Given a single frame I_t , we leverage the pixel-level dense motion information from optical flows to select dynamic-intensive tokens. First, we compute the optical flow between the current and the next frame, which contains not only subject motion information but also other global motions such as camera or background movement. To further eliminate interference from less informative content such as camera motion, we apply homography matrix compensation based on feature points extracted from the predicted flow. Subsequently, we leverage a salient detection mask to obtain the primary motion regions, enabling refined token selection. We then calculate the optical flow magnitude values (after camera motion filtering) for each pixel in these motion regions, select the tokens corresponding to pixels in the top $k\%$ magnitude values, and ultimately generate the final mask to identify valid tokens. In practice, we set k to 50. We utilize U2-Net [34] as the salient detection model, and adopt the powerful Sea-RAFT [44] model to extract accurate optical flows. The entire process can be formulated as,

$$f_t = \mathcal{E}(I_t, I_{t+1}), \quad (9)$$

$$f_t^* = f_t - \mathcal{H}(I_t, I_{t+1}, f_t), \quad (10)$$

$$m_t = \mathbb{I}(\|f_t^*\| \odot s_t \geq Q_{0.5}(\|f_t^*\| \odot s_t)), \quad (11)$$

$$q_t = p_t \odot m_t, \quad (12)$$

where f_t refers to the derived optical flow of I_t , and \mathcal{H} is the camera motion calculation function based on the homography matrix. \mathcal{E} refers to the optical flow network. s_t denotes the salient detection map of I_t and $\|f_t^*\|$ represents the magnitude of the filtered flow. \odot refers to the Hadamard product operator. $Q_{0.5}$ is the quantile function that select top 50% pixels with the highest dynamic degree and \mathbb{I} refers to the indicator function. p_t and q_t denote the original token and the filtered token from I_t , respectively. Here, we use fine-grained optical flow with more iterations to achieve precise pixel-level pruning.

When segmenting video events using coarse optical flow,

we ensure that each unit is assigned at least one frame to prevent information loss. After selecting key events based on semantic priors, we designate these key events along with their neighboring events as priority sampling events, where the number of sampled frames is proportional to the event length. Finally, within each priority-sampled event, we retain all tokens of the anchor frame to preserve complete contextual information, while applying the MTP to further refine intra-frame sampling for the adjacent frames.

4. Experiment

4.1. Implementation Details

Unless otherwise specified, our experiments are based on the LLaVA-Video-Qwen [56] extended with Flow4Agent. Results using other basic MLLMs can be found in the ablation studies. Following the standard settings, our image input resolution is 336, the LLM’s maximum context length is 8k, and the initial sampling frame count is 64. When performing intra-frame pruning, we simultaneously increase the sampling frame count to maintain the same visual context length as the original model. For motion priors, we use SeaRAFT [44] with 4 iterations in the TGO module and 12 iterations in the MTP module. For semantic priors, we reuse the base model’s encoder, SigLIP [51]. All experiments are conducted on two A100 GPUs. Implementation details of each basic model can be found in the appendix.

4.2. Benchmarks

We extensively tested the performance of Flow4Agent on six benchmarks, which primarily cover long video understanding and video reasoning. These benchmarks comprehensively evaluate whether our method can identify key video content from highly redundant information.

VideoMME [10] includes 900 videos of varying lengths and 2,700 manually annotated multiple-choice questions. The video lengths include short (<2min), medium (2-30min), and long (30-60min). Since the number of subtitles significantly impacts performance, we adopted a testing setup without subtitles.

LongVideoBench [46] is a dataset for long video retrieval and reasoning, containing 6,678 manually annotated multiple-choice questions and 17 fine-grained categories. The extraction of subtitles follows the official settings.

MLVU [57] is also a benchmark for long video understanding, containing nine different categories with video lengths ranging from 3 minutes to 2 hours, averaging 12 minutes.

Perception-Test [33] is a benchmark designed to test the perception and reasoning capabilities of MLLMs, containing 11.6k videos and six different annotation types.

EgoSchema [30] contains 5,000 video-question pairs for egocentric evaluation, each video lasting around 3 minutes.

NextQA [47] includes 5,440 videos and 49,000 questions,

Table 1. Flow4Agent performance on six video benchmarks, including NextQA, EgoChema, PerceptionTest, MLVU, LongVideoBench, and VideoMME. All results are reported as 0-shot accuracy.

Models	Size	NextQA	EgoSchema	PercepTest	MLVU	L-VideoBench	VideoMME	
							Long	Overall
Duration		44 sec	179.8 sec	16 sec	3~120 min	23sec~60 min	30~60 min	1~60 min
<i>Proprietary Models</i>								
GPT4-V [13]	-	-	55.6	-	-	59.1	56.9	60.7
<i>Open-Source Video MLLMs</i>								
Video-LLaVA [25]	7B	-	38.4	-	47.3	39.1	38.1	40.4
LLaMA-VID [21]	7B	-	38.5	-	33.2	-	-	-
ChatUniVi [15]	7B	-	-	-	-	-	41.8	45.9
ShareGPT4Video [3]	8B	-	-	-	46.4	39.7	37.9	43.6
LLaVA-NeXT-Video [23]	7B	70.2	43.9	59.4	39.3	50.5	-	46.5
VideoAgent [43]	7B	71.3	54.1	-	-	-	-	-
VideoTree [45]	7B	75.6	61.1	-	-	-	-	-
LVNet [32]	7B	72.9	61.1	-	-	-	-	-
VideoLLaMA2 [6]	7B	75.6	51.7	54.9	48.5	-	43.8	46.6
LongVA [54]	7B	69.3	-	-	56.3	-	47.6	54.3
VideoChat2 [20]	7B	-	54.4	-	47.9	36.0	39.2	54.6
LLaVA-OneVision [16]	7B	79.4	60.1	57.1	64.7	56.4	46.7	58.2
LLaVA-Video [56]	7B	83.2	57.3	67.9	70.8	58.2	50.6	62.6
Apollo [58]	7B	-	-	67.3	70.9	58.5	-	61.3
Flow4Agent	7B	84.0	61.4	69.6	71.4	60.4	54.2	64.7

primarily focusing on temporal, causal, and descriptive questions related to video understanding.

4.3. Main Results

Table 1 presents a quantitative comparison across multiple video understanding benchmarks. The experimental results show that Flow4Agent consistently outperforms previous state-of-the-art models on all benchmarks. For instance, compared to the latest model, Apollo, Flow4Agent demonstrates an advantage of 3.9%, 1.9%, and 3.4% on EgoSchema, LongVideoBench, and VideoMME, respectively. Notably, Flow4Agent performs particularly well on long videos. With similar frame sampling and context length, Flow4Agent outperforms LLaVA-Video and LLaVA-OneVision by 3.6% and 7.5%, respectively, on VideoMME videos longer than 30 minutes, and surpasses LongVA with a 224k context by 6.6%. This highlights Flow4Agent’s ability to extract key content from highly redundant videos within a limited context. On benchmarks emphasizing reasoning, such as PerceptionTest and EgoSchema, Flow4Agent also shows strong performance, despite the relatively short length of the videos. This suggests that extracting key information contributes to improving video reasoning capabilities. Furthermore, compared to other models focused on key video content extraction, such as VideoAgent, VideoTree, and LVNet, Flow4Agent still

shows a performance advantage, even though these models are based on the more powerful GPT-4. Interestingly, as a 7B model, Flow4Agent outperforms GPT-4V on most metrics. This demonstrates that motion priors can lead to improvements across different benchmarks.

4.4. Ablations and Analysis

Performance on Different Base Models. To validate the broad effectiveness of Flow4Agent, we conducted integration experiments with various base models. Table 2 presents the experimental results on VideoMME. We selected different types of models, including pure image models like LLaVA-Next, vision-general models like LLaVA-OneVision and Qwen2-VL, as well as pure video models like LLaVA-Video. The table also includes results with different context lengths, LLM sizes, and video frame counts. It is clearly observed that Flow4Agent consistently provides an improvement across various base models. Moreover, regardless of the underlying model, Flow4Agent shows the most significant improvement for long videos. This highlights the versatility of Flow4Agent as a model-agnostic method, particularly its ability to enhance long-form video understanding. Additionally, Flow4Agent maintains stable performance across different context lengths, LLM parameter sizes, and sampled frame numbers.

Effect of Different Components. Flow4Agent consists

Table 2. Flow4Agent performance on different basic models. We report the results on VideoMME without subtitles. All open-source results are our replication. We applied 4-bit quantization to LLaVA-Video-72B to ensure its deployment on two A100 GPUs.

Model	Context	LLM Params	Frames	Short	Medium	Long	Overall
LLaVA-NeXT [23]	4k	7B	16	54.3	41.9	38.4	44.9
LLaVA-NeXT + Flow4Agent	4k	7B	16	55.1	44.0	42.1	47.0
LLaVA-OneVision [16]	8k	7B	32	69.9	56.2	48.4	58.2
LLaVA-OneVision + Flow4Agent	8k	7B	32	70.9	57.3	51.6	59.9
Qwen2-VL [40]	32k	7B	64	73.0	60.8	51.3	61.7
Qwen2-VL + Flow4Agent	32k	7B	64	74.2	63.6	54	63.9
LLaVA-Video [56]	8k	7B	64	75.9	61.2	50.6	62.6
LLaVA-Video + Flow4Agent	8k	7B	64	77.2	62.6	54.2	64.7
LLaVA-Video	8k	72B	64	78.0	63.7	59.6	67.1
LLaVA-Video + Flow4Agent	8k	72B	64	80.1	66.9	61.6	69.0

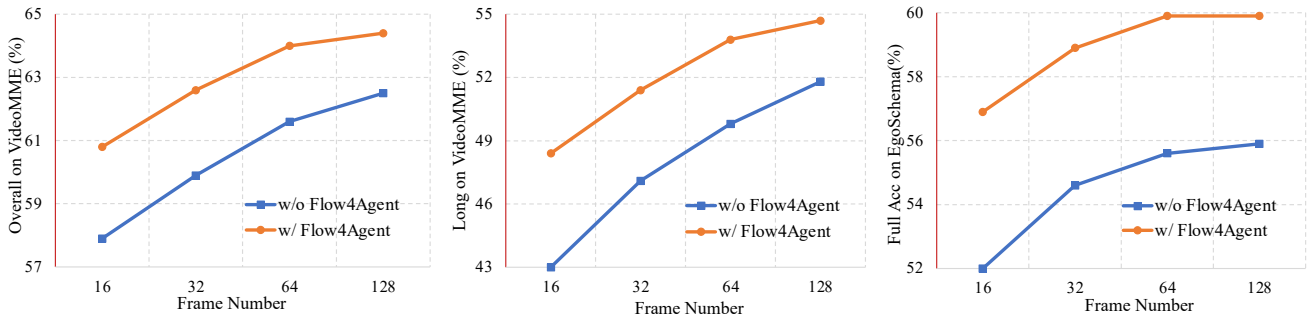


Figure 3. Performance comparison with and without Flow4Agent as the number of frames changes. Flow4Agent provides a greater improvement with fewer frames, while also achieving higher frame efficiency.

Table 3. The ablation study on model components. DES and ECCQ mean the dynamic event split and event-center cross-modal query respectively in the TGO module.

DES	ECQ	MTP	Short	Medium	Long	Overall
			75.9	61.2	50.6	62.6
✓			77.0	61.7	50.8	63.2
	✓		75.8	62.3	52.0	63.4
✓	✓		77.1	62.2	52.9	64.0
		✓	75.9	61.5	52.4	63.3
✓	✓	✓	77.2	62.6	54.2	64.7

of two core modules: TGO and MTP, with TGO further divided into motion-guided event splitting and semantic-guided event selection. To assess the impact of these modules, we conducted ablation experiments on the overall model components. In the TGO module, when we use only the motion prior (DES only), frames are allocated to all events directly based on their length. When we use only the semantic prior (ECQ only), semantic checks are performed on individual frames rather than events. As shown in Table 3, the two components within TGO provide significant gains for short and long videos, respectively, and their combination leads to improved performance across all video

lengths. The MTP module further enhances long video understanding. Each module complements the others, collectively demonstrating the design of Flow4Agent.

Effect of Different Frames Number. The number of frames is a crucial variable affecting video understanding, particularly for long videos. In theory, a sufficient number of frames ensures comprehensive coverage of necessary information to answer a given question. However, an excessive number of frames can introduce redundant or irrelevant information, potentially overwhelming the model. Thus, the ability to extract key information within a constrained frame budget is a critical metric for evaluating model performance. As shown in Fig. 3, we analyzed performance variations across different frame counts using three test sets with varying video lengths: VideoMME-Overall, VideoMME-Long, and EgoSchema. To support a 128-frame input, we adjusted the avgpooling2d kernel size from 2 to 3. The results indicate that Flow4Agent consistently enhances performance regardless of the number of input frames. When frame availability is limited, Flow4Agent’s advantage becomes even more pronounced. Additionally, Flow4Agent achieves performance saturation with fewer frames, demonstrating higher frame efficiency. These findings highlight Flow4Agent’s effectiveness in extracting critical video in-

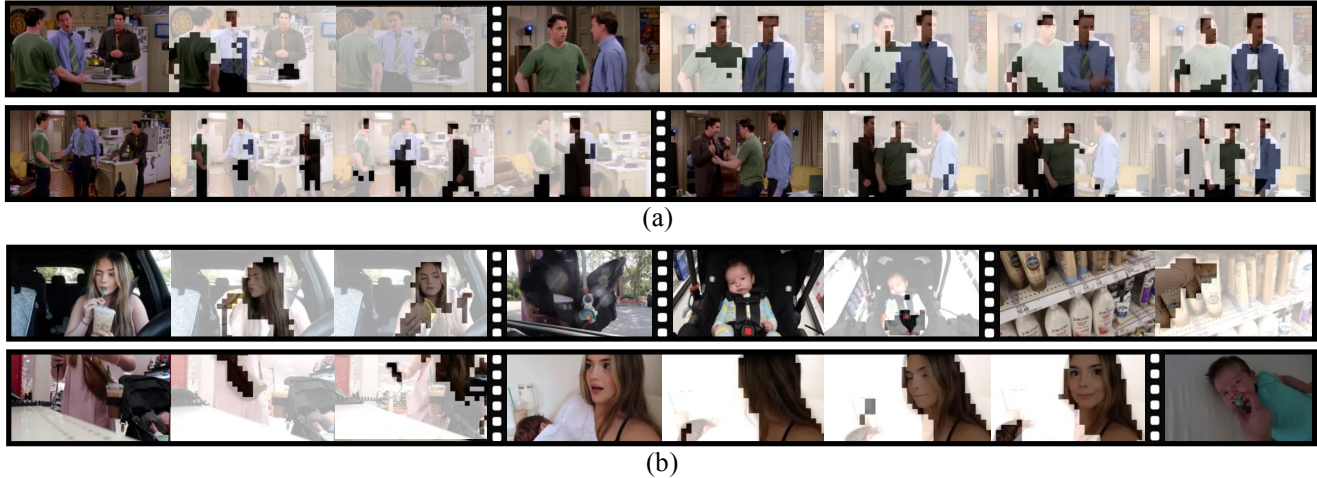


Figure 4. Visualizations of how Flow4Agent reduces redundancy.

Table 4. The ablation study on the motion-prior model. Iter-TGO and Iter-MTP refer to the number of iterations of the optical flow model within the TGO and MTP modules, respectively.

Flow Model	Iter-TGO	Iter-MTP	Long	Overall
NeuFlow	4	12	53.0	64.1
StreamFlow	4	12	53.9	64.5
Sea-RAFT	4	12	54.2	64.7
Sea-RAFT	12	12	54.4	64.6
Sea-RAFT	4	4	53.3	64.2

Table 5. The ablation study on semantic-prior model.

CLIP Model	Size	Resolution	Long	Overall
OpenAI-CLIP	0.4B	224	53.3	63.9
EVA-CLIP	8B	224	53.1	64.0
SigLIP	0.4B	224	53.9	64.2
SigLIP	0.4B	336	54.2	64.7

formation while optimizing frame utilization.

Effect of the Prior Model. Motion and semantics are two essential priors leveraged by Flow4Agent, and the choice of corresponding models as well as their configurations can significantly influence the final performance. In Table 4, we examine different optical flow models and their iteration counts within the TGO and MTP modules. While more iterations yield finer optical flow information, they also increase computational time. Our results show that the state-of-the-art Sea-RAFT model delivers superior performance. Additionally, in the TGO module, fewer iterations can obtain optimal results, whereas the MTP module benefits from more iterations for the best performance. This highlights the optimal configuration of Flow4Agent: coarse optical flow for event splitting and fine-grained optical flow for visual token pruning. In Table 5, we explore various semantic prior models. The SigLIP-336 model achieves the best re-

sults, demonstrating that stronger semantic priors contribute to improved performance.

4.5. Visualization

To qualitatively assess the effectiveness of Flow4Agent, we present several visualization cases in Fig. 4. Different scenes identified by the TGO module within the same video are separated by film lines, while redundant regions filtered out by the MTP module within each scene are grayed out. Across all cases, we observe that the TGO module effectively differentiates distinct scenes. For instance, in Fig. 4(a), despite the highly similar background, the TGO module successfully distinguishes between a two-person conversation and a three-person group interaction. Additionally, the MTP module efficiently removes redundant background elements within the same scene while preserving crucial variations, such as human actions and facial expressions. More examples can be found in the appendix.

5. Conclusion

In this paper, we propose Flow4Agent, which introduces optical flow into LLM-based video understanding for the first time, incorporating a novel motion prior to extract key video information. Flow4Agent enhances long-form video comprehension through two modules: the Temporal Granularity Optimization (TGO) module that leverages coarse motion and semantic information to eliminate inter-frame redundancy and identify key events, and the Motion Token Pruning (MTP) module that utilizes fine-grained optical flow to remove intra-frame redundancy. Extensive quantitative and ablation experiments demonstrate the effectiveness of Flow4Agent in long-form video understanding and video reasoning, achieving state-of-the-art performance across a wide range of video benchmarks.

Acknowledgements. This work was supported by National Science and Technology Major Project (2024ZD01NL00101).

References

- [1] Daechul Ahn, Yura Choi, Youngjae Yu, Dongyeop Kang, and Jonghyun Choi. Tuning large multimodal models for videos using reinforcement learning from ai feedback. *arXiv preprint arXiv:2402.03746*, 2024. 2
- [2] Meng Cao, Haoran Tang, Haoze Zhao, Hangyu Guo, Jiaheng Liu, Ge Zhang, Ruyang Liu, Qiang Sun, Ian Reid, and Xiaodan Liang. Physgame: Uncovering physical commonsense violations in gameplay videos. *arXiv preprint arXiv:2412.01800*, 2024. 2
- [3] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, et al. Sharegpt4video: Improving video understanding and generation with better captions. *arXiv preprint arXiv:2406.04325*, 2024. 6
- [4] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, et al. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13320–13331, 2024. 3
- [5] 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. 1
- [6] 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. 6
- [7] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. See <https://vicuna.lmsys.org> (accessed 14 April 2023), 2(3):6, 2023. 2
- [8] W Dai, J Li, D Li, AMH Tiong, J Zhao, W Wang, B Li, P Fung, and S Hoi. Instructblip: towards general-purpose vision-language models with instruction tuning. *arXiv Preprint posted online on June, 15:2023*, 2023. 2
- [9] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 1
- [10] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024. 2, 5
- [11] Qifan Fu, Yichun Zhang, Liyong Xu, and Huixin Li. A method of shot-boundary detection based on hsv space. In *2013 Ninth International Conference on Computational Intelligence and Security*, pages 219–223. IEEE, 2013. 3
- [12] Kai Han, Jianyuan Guo, Yehui Tang, Wei He, Enhua Wu, and Yunhe Wang. Free video-llm: Prompt-guided visual perception for efficient training-free video llms. *arXiv preprint arXiv:2410.10441*, 2024. 1, 3, 5
- [13] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4v system card. *arXiv preprint arXiv:2410.21276*, 2024. 6
- [14] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 1
- [15] Peng Jin, Ryuichi Takanobu, Wancai Zhang, Xiaochun Cao, and Li Yuan. 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*, pages 13700–13710, 2024. 1, 2, 3, 6
- [16] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 1, 2, 3, 6, 7
- [17] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024. 2
- [18] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2
- [19] KunChang Li, Yanan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 2
- [20] Kunchang Li, Yali Wang, Yanan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. *arXiv preprint arXiv:2311.17005*, 2023. 2, 6
- [21] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pages 323–340. Springer, 2024. 1, 2, 3, 6
- [22] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2
- [23] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 2, 6, 7
- [24] Jiajun Liu, Yibing Wang, Hanghang Ma, Xiaoping Wu, Xiaoyi Ma, Xiaoming Wei, Jianbin Jiao, Enhua Wu, and Jie Hu. Kangaroo: A powerful video-language model supporting long-context video input. *arXiv preprint arXiv:2408.15542*, 2024. 1, 3

- [25] Ruyang Liu, Chen Li, Yixiao Ge, Thomas H Li, Ying Shan, and Ge Li. Bt-adapter: Video conversation is feasible without video instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13658–13667, 2024. 2, 6
- [26] Ruyang Liu, Chen Li, Haoran Tang, Yixiao Ge, Ying Shan, and Ge Li. St-llm: Large language models are effective temporal learners. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024. 2, 3
- [27] Ruyang Liu, Haoran Tang, Haibo Liu, Yixiao Ge, Ying Shan, Chen Li, and Jiankun Yang. Ppllava: Varied video sequence understanding with prompt guidance. *arXiv preprint arXiv:2411.02327*, 2024. 1, 3, 5
- [28] Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Minghui Qiu, Pengcheng Lu, Tao Wang, and Zhongyu Wei. Valley: Video assistant with large language model enhanced ability. *arXiv preprint arXiv:2306.07207*, 2023. 2
- [29] 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 (ACL 2024)*, 2024. 2
- [30] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 5
- [31] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, 2024. 5
- [32] Jongwoo Park, Kanchana Ranasinghe, Kumara Kahatapitiya, Wonjeong Ryoo, Donghyun Kim, and Michael S Ryoo. Too many frames, not all useful: Efficient strategies for long-form video qa. *arXiv preprint arXiv:2406.09396*, 2024. 1, 3, 6
- [33] Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Mateusz Malinowski, Yi Yang, Carl Doersch, et al. Perception test: A diagnostic benchmark for multimodal video models. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 5
- [34] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern recognition*, 106:107404, 2020. 5
- [35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2
- [36] Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, et al. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*, 2024. 3, 5
- [37] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014. 3
- [38] Enxin Song, Wenhao Chai, Guan hong 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. 1, 3
- [39] Shuyang Sun, Zhanghui Kuang, Lu Sheng, Wanli Ouyang, and Wei Zhang. Optical flow guided feature: A fast and robust motion representation for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1390–1399, 2018. 3
- [40] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 1, 7
- [41] Xijun Wang, Junbang Liang, Chun-Kai Wang, Kenan Deng, Yu Lou, Ming C Lin, and Shan Yang. Vila: Efficient video-language alignment for video question answering. In *European Conference on Computer Vision*, pages 186–204. Springer, 2024. 2
- [42] Xiao Wang, Qingyi Si, Jianlong Wu, Shiyu Zhu, Li Cao, and Liqiang Nie. Retake: Reducing temporal and knowledge redundancy for long video understanding. *arXiv preprint arXiv:2412.20504*, 2024. 1, 3
- [43] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pages 58–76. Springer, 2024. 1, 3, 6
- [44] Yihan Wang, Lahav Lipson, and Jia Deng. Sea-raft: Simple, efficient, accurate raft for optical flow. In *European Conference on Computer Vision*, pages 36–54. Springer, 2024. 3, 5
- [45] Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. *arXiv preprint arXiv:2405.19209*, 2024. 1, 3, 6
- [46] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2025. 2, 5
- [47] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9777–9786, 2021. 2, 5
- [48] Lin Xu, Yilin Zhao, Daquan Zhou, Zhijie Lin, See Kiong Ng, and Jiashi Feng. Pllava: Parameter-free llava extension from images to videos for video dense captioning. *arXiv preprint arXiv:2404.16994*, 2024. 1, 2, 3
- [49] Mingze Xu, Mingfei Gao, Zhe Gan, Hong-You Chen, Zhengfeng Lai, Haiming Gang, Kai Kang, and Afshin

- Dehghan. Slowfast-llava: A strong training-free baseline for video large language models. *arXiv preprint arXiv:2407.15841*, 2024. 3
- [50] 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. 1, 2
- [51] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 2, 5
- [52] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 2
- [53] Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, Jifeng Dai, and Xiaojie Jin. Flash-vstream: Memory-based real-time understanding for long video streams. *arXiv preprint arXiv:2406.08085*, 2024. 1, 3
- [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. 1, 3, 6
- [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. 2
- [56] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024. 2, 3, 5, 6, 7
- [57] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024. 2, 5
- [58] Orr Zohar, Xiaohan Wang, Yann Dubois, Nikhil Mehta, Tong Xiao, Philippe Hansen-Estruch, Licheng Yu, Xiaofang Wang, Felix Juefei-Xu, Ning Zhang, et al. Apollo: An exploration of video understanding in large multimodal models. *arXiv preprint arXiv:2412.10360*, 2024. 2, 3, 6