DIvide, then Ground: Tailoring Frame Selection to Query Types for Long-Form Video Understanding

Anonymous Author(s)
Affiliation
Address
email

Abstract

The effectiveness of Large multimodal models (LMMs) in long-form video understanding is constrained by limited context length. Whether conventional approaches, such as uniform frame sampling, can adequately address this limitation remains uncertain. In this paper, we observe that increasing the number of uniformly sampled input frames doesn't consistently enhance model performance. Further analysis reveals that this issue is due to the query type, with performance degrading mainly on localized queries. To address this, we propose **DIG**, a training-free frame selection framework for LMMs that adapts to query type by employing uniform sampling for global queries and a specialized pipeline to extract query-relevant frames for localized queries. Experimental results demonstrate that **DIG** consistently improves performance across multiple benchmarks, including MLVU, LongVideoBench, and VideoMME, when applied to different LMMs and under varying input frames counts. Notably, with 320 input frames, **DIG** achieves accuracy improvements of 2.71% on MLVU and 2.32% on LongVideoBench.

1 Introduction

In recent years, there has been a rapid advancement in large multimodal models (LMMs) [23, 42, 19, 36, 9] for open-world visual understanding. A natural and increasingly important direction within this field is the extension of these models to handle video data, thereby enabling them to perform complex video understanding tasks [26, 22, 58, 21, 17, 33, 57, 10, 62]. The common approach [59, 3] involves representing videos as sequences of individual frames, where visual features are extracted from each frame and concatenated to form a video representation that is subsequently processed by the large language model (LLM). However, due to the limited context length of the LLM and the sheer volume of video tokens, it is impractical to input all frames directly. As a result, only a sampled subset of frames is typically used as input. This constrained sampling strategy inevitably leads to incomplete coverage of the video content, thereby causing substantial information loss and impairing the model's holistic understanding of the video.

To address this limitation, recent approaches have focused on two main strategies: extending the context length of the models to accommodate more frames [38, 8], or developing frame selection mechanisms that identify and utilize the most representative frames based on the given query [25, 41, 56, 55, 40]. However, our findings indicate that simply increasing the number of uniformly sampled input frames does not consistently enhance model performance and may lead to a degradation in results when an excessive number of frames is employed. To investigate this, we classify queries into two types: *localized queries*, which pertain to specific segments within the video, and *global queries*, which necessitate comprehension of the entire video content. Our experiment reveals that the observed decline in performance is primarily attributable to localized queries. This is because incorporating additional frames for such queries tends to introduce irrelevant or noisy information,

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

whereas global queries benefit from broad frame coverage. Based on this analysis, it is evident that the careful selection of only the most relevant frames for localized queries plays a critical role in enhancing overall performance. In contrast, uniform sampling works well for global queries, and applying query-based frame selection strategies in these cases may degrade performance.

Building on these findings, we propose **DIG**, a training-free frame selection framework for LMMs that adapts to the type of query. Specifically, given a query, the LMM first classifies it as either global or localized. For global queries, uniform sampling is employed to ensure comprehensive coverage. In contrast, for localized queries, we introduce a carefully designed pipeline:

We first introduce *Content-Adaptive Frame Selection*, a method that selects a set of representative frames (*r-frames*) capable of capturing the semantic content and major visual elements of the entire video. This approach leverages pairwise frame similarity computations based on DINO features [29] to identify the most informative frames. The LMM itself is then employed to assign a reward to each *r-frame* by evaluating its direct usefulness or determining whether adjacent *r-frames* contain relevant information that contributes to answering the query. Guided by this reward distribution, we design a video refinement process that adaptively selects *r-frames* and merges the video segments they represent into a refined, condensed video. In this refined video, where irrelevant content has been filtered out, uniformly sampled frames from the remaining segments are used as input to the LMM for the final inference.

Our main contribution are summarized as follows:

- We demonstrate that including an excessive number of uniformly sampled frames can degrade performance on localized queries, while such an approach performs well for global queries.
- We propose **DIG**, a training-free frame selection framework for LMMs that adapts to query type by employing uniform sampling for global queries and a specialized pipeline to extract query-relevant frames for localized queries.
- Extensive experiments shows **DIG** improve LMM's performance consistently on three benchmarks, including a 2.71% gain on MLVU[65] and a 2.32% improvement on LongVideoBench [49].

2 Related Work

2.1 Video-based Large Multimodal Models

The rise of Transformer-based large language models (LLMs) has revolutionized natural language processing, with major advances stemming from increased model scale and larger pre-training datasets [13, 28, 4, 43, 30, 11, 12, 61]. Inspired by this success, researchers have begun adapting LLMs to process multiple modalities, particularly integrating visual elements like images and videos [66, 19, 23, 24], leading to the development of LMMs. Through extensive training, these models learn rich, cross-modal representations that effectively connect visual and textual information. This evolution has led to significant improvements across a range of video understanding applications, including tasks such as video captioning [53, 7, 50, 5, 52] and video question answering [18, 27, 26, 10, 57, 64]. Ongoing research is also focusing on refining model architectures [37, 51] and optimizing training strategies [68, 24] to further boost the performance of these multimodal systems.

Despite their success, LMMs still struggle in video understanding due to the high volume of video tokens and the limited context length of LLMs [38, 8], as well as the "Needle-in-a-Haystack" issue [63, 55, 20]. These challenges highlight the need for efficient frame selection techniques that capture key visual content without overloading the model.

2.2 Video Token Reduction in LMMs for Video Question Answering (VQA)

In VQA task, uniform frame sampling is commonly used for video token reduction but it fails to account for varying frame relevance to the query. To overcome this limitation, recent approaches have focused on making token reduction more adaptive, and can be broadly categorized into two types.

Token Compression. This strategy consolidates information within or across video frames to create a more compact yet informative representation, reducing the total tokens needed. Various techniques are employed to achieve this, such as using a memory bank [39], reducing temporal redundancy [34, 45], and applying hierarchical compression [20]. Despite their efficiency, token

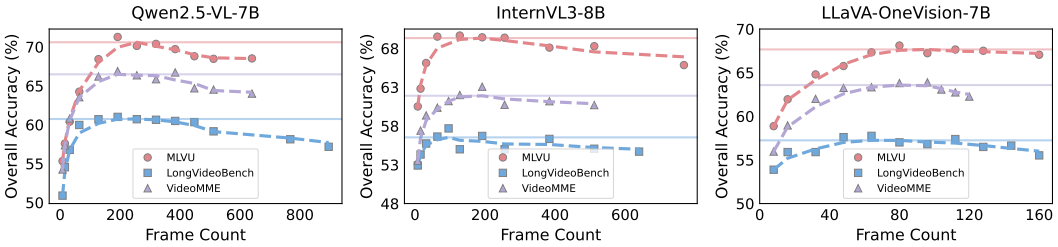

Figure 1: Accuracy of various models across three benchmarks, evaluated with varying input frames.

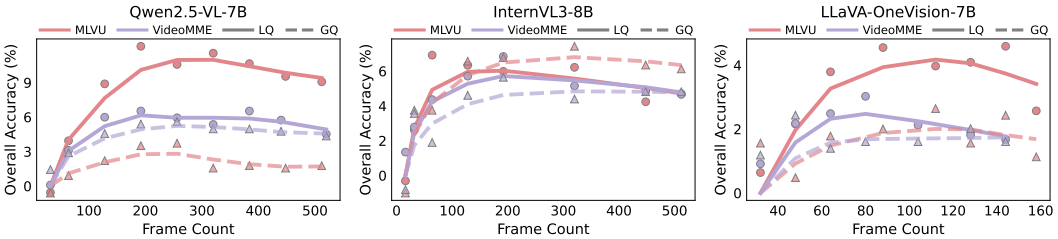

Figure 2: Accuary of various model across MLVU [65] and VideoMME [15] on localized queries and global queries, with the difference from the initial point recorded.

compression techniques may lead to excessive summarization, resulting in the loss of fine-grained visual details. Moreover, query-related segments may be either compressed or overly generalized, potentially compromising the model's capacity to effectively respond to the given query.

Query-Based Frame Selection. To address the limitations of uniform sampling, recent methods employ more refined strategies to select query-relevant frames that typically involve three key steps: (1) uniformly sample candidate frames or segments [25, 41, 40] or video segments [2]; (2) assess their relevance to the query using metrics like CLIPScore [16] or learned models [56]; (3) apply an algorithm to select the most relevant frames based on these scores. However, these methods treat all queries the same, but for some queries, most frames contribute equally, making frame selection potentially detrimental. In this work, we discuss this phenomenon and propose a training-free frame selection strategy to enhance the performance of LMMs at inference time.

3 Revisiting Inference Mechanism of LMM in Video Understanding

Consider a video V with T frames, denoted as $\{f_i\}_{i=1}^{T}$, along with a query Q. Among various tasks in video understanding, Video Question Answering (VQA) is a particularly prominent one. In this task, the model receives with the video V and the query Q as inputs and is tasked with generating a response A that accurately addresses the query. In contemporary approaches, LMMs address this challenge through the following process. Due to computational limitations and the language model's restricted context length L, only a subset of N uniformly sampled frames, denoted as $\{f'_i\}_{i=1}^{N}$, is processed, where $N \ll T$. These selected frames are then combined with the query Q and fed into the LMM, which autoregressively generates the answer A:

$$A = \text{LMM}([f'_1; f'_2; \ldots; f'_N; Q]). \tag{1}$$

Obviously, a small subset of N frames is often insufficient to capture the full content of a video, particularly in longer sequences. To address this, recent studies [8, 60] have focused on extending model context lengths to allow more frames as input. However, this raises an important question: *does increasing the number of uniformly sampled input frames enhance performance on VQA task?*

Performance Decline with Excessive Frames. To investigate this, we conducted an evaluation using three pretrained LMMs: Qwen2.5-VL-7B [3], InternVL3-8B [67], and LLaVA-OneVision-7B [19], across three long-form video understanding benchmarks: MLVU [65], VideoMME [15], and LongVideoBench [49]. We employed uniform frame sampling with varying frame counts to evaluate the impact of frame count on model performance. As illustrated in Figure 1, a consistent pattern emerges across all models and benchmarks: performance initially improves with more input frames but declines beyond a certain point.

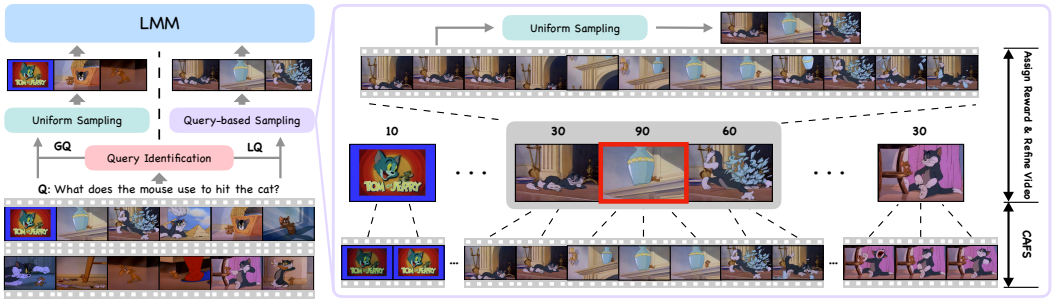

Figure 3: **Overview of DIG.** When the query is identified as global, frames are uniformly sampled from the entire video. For localized queries, the video is processed through CAFS and reward assignment to construct a refined video, from which frames are then uniformly sampled.

Query Classification. To better understand the performance degradation, we examined the impact of different query types. Prior studies [25, 41, 35, 6, 31] have identified a class of queries that relate to specific, localized segments of a video, such as *"What kind of vehicle is the man riding?"*, which we classify as *localized queries (LQ)*. However, these works overlook another important category of queries that require a comprehensive understanding of the entire video content. We define such queries as *global queries (GQ)*, with an example being *"What title best summarizes this video?"*

Performance Trends Vary across Query Types. Following the definition, we categorize queries from MLVU [65] and VideoMME [15], and evaluate the same models on these two query types. LongVideoBench[49] contains only localized queries due to its design, so its results align with those in Figure 1. As shown in Figure 2, while performance on global queries remains relatively stable with increasing frame count, performance on localized queries drops significantly. This is because global queries benefit from information across most frames, whereas localized queries rely on only a specific subset of frames. Including additional frames may introduce noise, impairing the model's ability to extract the key information. This underscores the importance of employing distinct frame selection strategies tailored to different query types.

4 Method

In this section, we introduce **DIG**, a training-free frame selection framework for LMMs that adapts to query type. **DIG** begins by using the LMM to classify the query as localized or global (§4.1). For global queries, final input frames are uniformly sampled across the entire video. In contrast, for localized queries, we employ a content-adaptive frame selection method to extract representative frames (§4.2), which are then evaluated by the LMM through reward scoring to assess their relevance to the query (§4.3). Then a refined video is constructed through a search procedure guided by these rewards (§4.4) and final input frames are uniformly sampled from the refined video.

4.1 Query Type Identification

Following the discussion in Section 3, it is crucial to adopt query-specific frame selection strategies. Therefore, we first let the LMM classify the given query Q as global or localized, using a simplified prompt illustrated here. For global queries, the LMM performs direct inference on uniformly sampled frames. For localized queries, we apply the specialized approach detailed in the following sections.

Query Identification Prompt (Simplified)

You are a helpful assistant in a video-based question-answering task. Your role is to determine whether the given query requires understanding of the entire video or can be answered by analyzing specific segments, based on contextual cues in the query.
Query: $< Q >$; Provide a brief analysis and then make your judgment.

4.2 Content-Adaptive Frame Selection (CAFS)

To effectively address the localized query, it is essential to extract relevant information from the video. However, processing long videos is computationally intensive due to temporal redundancy.

To address this, we introduce *Content-Adaptive Frame Selection (CAFS)*, a method that adaptively selects representative frames, referred to as *r-frame*, based on the high-level semantic content in the video, such as objects and scenes. This approach ensures that the *r-frames* form a compact yet informative summary of the video.

Distance Calculation. Given a 2-fps sampled video with M frames $\{f_{I_i}\}_{i=1}^{M}$ with corresponding frame indices $\{I_i\}_{i=1}^{M}$, we utilize DINOv2 [29] to extract visual features from each frame, which results in a sequence of feature vectors $\{V_{I_i}\}_{i=1}^{M}$. To measure the dissimilarity between consecutive frames, we compute the feature distance d_i between f_{I_i} and $f_{I_{i+1}}$ using the following formula:

$$d_i = 1 - \text{sim}(V_{I_i}, V_{I_{i+1}}), \tag{2}$$

where $\text{sim}(\cdot, \cdot)$ denotes cosine similarity. This yields a sequence of distances $\{d_i\}_{i=1}^{M-1}$.

R-Frame Selection. Due to frequent scene transitions or camera cuts in long videos, the pairwise frame similarity often exhibits abrupt changes, resulting in numerous peaks in the distance sequence. Specifically, d_i is identified as a peak if $d_{i-1} < d_i$ and $d_{i+1} < d_i$. To reduce noise effects, only peaks with prominence greater than 0.1 are valid. This threshold has been found effective through empirical observation. We denote the indices of these valid peaks as $\{K_j\}_{j=1}^{N} \subset \{I_i\}_{i=1}^{M}$, where $N < M$. These peaks serve as segmentation points, dividing the video into distinct segments. Within each segment, the low pairwise distances between frames indicate visual consistency. Therefore, we select only one frame from each segment to capture its semantic content. For simplicity, we choose the midpoint frame of each segment, resulting in a set of *r-frames* indexed by $\{I'_j\}_{j=1}^{N-1} = \{(K_j + K_{j+1})/2\}_{j=1}^{N-1}$. By aggregating *r-frames*, we obtain a compact representation that effectively summarizes the essential visual content of the entire video.

Algorithm 1 Content-Adaptive Frame Selection

Input: Distance sequence $d = [d_1, d_2, \ldots, d_{M-1}]$, frame indices $I = [I_1, I_2, \ldots, I_M]$
`peaks` $\leftarrow \varnothing$
for $i = 2, \ldots, M-1$ **do**
  Add i to `peaks` if $d_{i-1} < d_i < d_{i+1}$
end for
for $j \in$ `peaks` **do**
  $l_{\text{min}} \leftarrow d_j, k \leftarrow j - 1$
  while $k \geq 0$ and $d_k \leq d_j$ **do**
    $l_{\text{min}} \leftarrow d_k$ if $d_k < l_{\text{min}}$
    $k \leftarrow k - 1$
  end while
  $r_{\text{min}} \leftarrow d_j, m \leftarrow j + 1$
  while $m \leq N$ and $d_m \leq d_j$ **do**
    $r_{\text{min}} \leftarrow d_m$ if $d_m < r_{\text{min}}$
    $m \leftarrow m + 1$
  end while
  `prominence` $\leftarrow d_j - \max(l_{\text{min}}, r_{\text{min}})$
  Remove j from `peaks` if `prominence` ≤ 0.1
end for
`idx` $\leftarrow \{(I_{\texttt{peaks[i]}} + I_{\texttt{peaks[i+1]}})/2\}_{i=0}^{\text{len(peaks)}}$
return `idx`

4.3 Reward Assignment

To identify *r-frames*'s relevance to the given query Q, existing methods typically use either: (1) multimodal models like CLIP [32] to align text and visual embeddings [25, 41, 46], or (2) object detection models to localize query-related entities in frames [55]. However, these methods are often constrained by surface-level feature matching and reliance on fixed vocabularies, which limits their ability to capture contextual reasoning and broader world knowledge. To address this, we leverage the LMM itself to assess frame relevance by assigning reward scores.

Independent Frame Scoring. The LMM is employed as a reward model to evaluate each *r-frame* individually, assigning a reward score that quantifies the *r-frame*'s relevance to the given query Q. To enhance interpretability, we incorporate the chain-of-thought reasoning technique [48], prompting the model to generate a description for each frame before producing the reward.

Two-Dimension Scoring. Since many queries, particularly those involving "why" or "how", cannot be fully addressed by a single frame, evaluating the relevance of individual frames individually may lead to incomplete or biased assessments. To mitigate this, we design the LMM to consider two complementary factors: (1) the direct relevance of the current frame to the query, and (2) whether the content of the current frame indicates that adjacent frames may contain supplementary information that contributes to a more comprehensive response.

Let $\{f_i\}_{i=1}^{N}$ denote the *r-frames* extracted by CAFS. A simplified version of the prompting strategy is illustrated here. It is worth noting that, for LMM, this serves as a clean and efficient framework, guided solely by its self-generated intrinsic reward and requiring no external training.

Reward Model Prompt (Simplified)

You are acting as a reward model to guide a video-based question-answering process.
Frame: $< f_i >$; Query: $< Q >$; Please follow these steps to finish scoring:
1. Describe the sampled frame, focusing only on elements relevant to the question, if any.
2. Assign a relevance score between 0 and 100 based on: (1) Direct usefulness of the frame for answering the query. (2) Whether it suggests adjacent frames may contain relevant context.

4.4 Video Refinement

Building upon the preceding steps, we have obtained the set of peak indices $\{K_j\}_{j=1}^{N}$, the *r-frame* indices $\{I'_j\}_{j=1}^{N-1}$, and the reward values $\{R_j\}_{j=1}^{N-1}$ assigned to these *r-frames*. The next step is to select the most query-relevant *r-frames* based on the reward values $\{R_j\}_{j=1}^{N-1}$.

Iterative Reward-Based Selection. In contrast to the commonly employed Top-K selection, which applies a fixed hyperparameter across varying scenarios, we introduce a parameter-free methodology. Given the initial set of rewards $\{R_j\}_{j=1}^{N-1}$, we iteratively refine this set until it stabilizes.

- *Step 1.* Compute the mean of the current reward set: $\overline{R}$.
- *Step 2.* Update each reward value by thresholding below the mean:

$$R'_j = \max(R_j - \overline{R}, 0), \quad \forall j = 1, \ldots, N-1. \tag{3}$$

- *Step 3.* Let S be the set of indices $\{j \mid R'_j > 0\}$. Compare S with the set of positive indices from the previous iteration. If S is unchanged, terminate the iteration. Otherwise, update the reward set $\{R_j\} \leftarrow \{R'_j\}$ and repeat from *Step 1*.

Upon termination, the selected *r-frames*, denoted by I_f, are formally defined as those *r-frames* whose corresponding reward values in the final iteration are positive: $I_f = \{I'_j \mid R'_j > 0\}_{j=1}^{N-1}$. This criterion ensures that all *r-frames* in the final selection set possess a reward larger than average.

Segment Combination. Since *r-frames* exhibit high feature similarity with their adjacent frames, it indicates an opportunity to incorporate fine-grained information beyond simply using them as input to the LMM. Specifically, for each selected *r-frame* indexed by I'_j, we consider the video segment in the interval $[K_j, K_{j+1}]$ for richer temporal details. In addition, to capture more relevant context, we also consider adjacent *r-frames* within a window of length $wlen$, specifically those with index range from I'_{j-wlen} to I'_{j+wlen}. This results in the video segment spanning the index range $[K_{j-wlen}, K_{j+wlen+1}]$. Then we combine the corresponding video segments of all selected *r-frames* via a union operation, resulting in a refined video containing query-relevant and fine-grained content. Finally, we uniformly sample frames from this refined video for input to the LMM.

5 Experiment

5.1 Experiment Settings

Datasets. We evaluate our approach on three benchmarks: MLVU [65], LongVideoBench [49], and VideoMME [15]. These datasets contain videos ranging from several minutes to multiple hours, allowing us to assess long-form video understanding. For VideoMME [15], we focus only on the medium and long splits. We don't use subtitles, ensuring evaluation is based on visual understanding.

Implementation Details. The LMMs used in our experiments are Qwen2.5-VL-7B[3] and Qwen2-VL-7B [44]. Each image is represented using 256 tokens. The hyperparameter $wlen$ is set to 2. The comparison is between the incorporation of **DIG** and uniform sampling when doing inference. All experiments are conducted on four A100 GPUs.

5.2 Results

Comparison with Uniform Sampling. As shown in Figure 4, compared with uniform sampling across the entire video, **DIG** consistently improves performance on both Qwen2.5-VL-7B [3] and Qwen2-VL-7B [44] across input frame numbers from 8 to 320. Notably, with 320 frames, **DIG** boosts

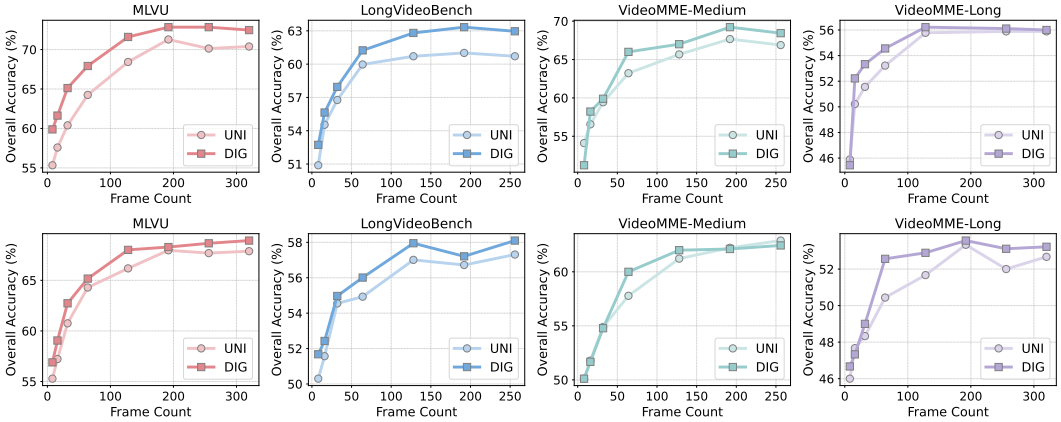

Figure 4: Comparison between **DIG** and uniform sampling (UNI). The top four charts are based on Qwen2.5-VL-7B[3], while the bottom four are based on Qwen2-VL-7B [44].

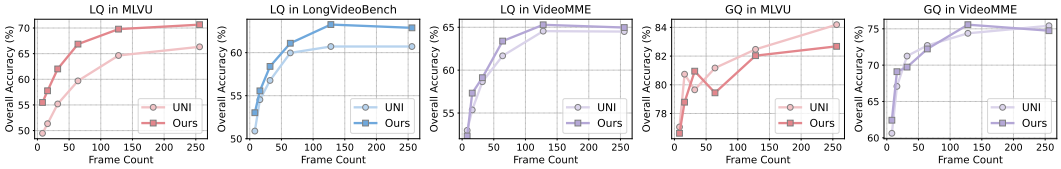

Figure 5: Comparison between uniform sampling (UNI) and the proposed pipeline, as described from Section4.2 to Section 4.4, across query types. The evaluation is conducted by Qwen2.5-VL-7B [3].

the accuracy of Qwen2.5-VL-7B by 2.71% on MLVU [65] and by 2.32% on LongVideoBench [49]. These results underscore the significance of filtering query-relevant frames during inference to fully exploit the capabilities of existing LMMs when processing localized queries.

Comparison on Different LMMs. Notably, the performance gain is more significant on Qwen2.5-VL-7B [3] than on Qwen2-VL-7B [44]. Specifically, for Qwen2.5-VL-7B [3], the accuracy improves by an average of 3.49% on MLVU [65], while for Qwen2-VL-7B [44], the improvement is only 1.31%. On LongVideoBench [49], the improvements are 1.40% and 0.85%, respectively. This is because a stronger LMM better identifies query types and assigns suitable rewards, resulting in greater gains (§5.3 and §5.5), highlighting **DIG**'s potential when paired with more powerful models.

5.3 Ablation Study: Query Identification

We investigate how different frame selection strategies affect performance on global and localized queries. Utilizing the query classification results mentioned in Section 3 on MLVU [65], LongVideoBench [49] and VideoMME [15], we compare two methods on each query type: uniform sampling across the entire video, and the pipeline described from Section 4.2 to Section 4.4.

GQ Benefits from Uniform Sampling. As shown in the right two line charts in Figure 5, our pipeline performs similarly or worse than uniform sampling on global queries. This is expected, as frames selected through uniform sampling can cover a broader range of the video content, while our pipeline targets more specific segments, which are less likely to capture diverse and rich information.

LQ benefits from DIG. As shown in the left three line charts in Figure 5, our pipeline significantly outperforms uniform sampling on localized queries, demonstrating its effectiveness in handling such queries by accurately extracting relevant video segments. These results highlight one key takeaway: when processing a query, it's essential to first identify the query type, and then decide whether to extract information from the entire video or to first extract relevant segments before analyzing them.

5.4 Ablation Study: CAFS

We evaluate CAFS from two aspects: (1) the capability of CAFS to extract frames which represents high-level semantic video content, and (2) the performance of CAFS when integrating in **DIG**.

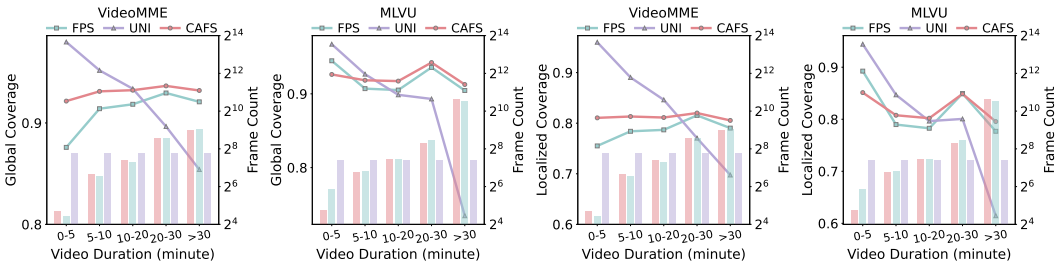

Figure 6: LoC and GlC of fps sampling, uniform sampling and CAFS across varying video length.

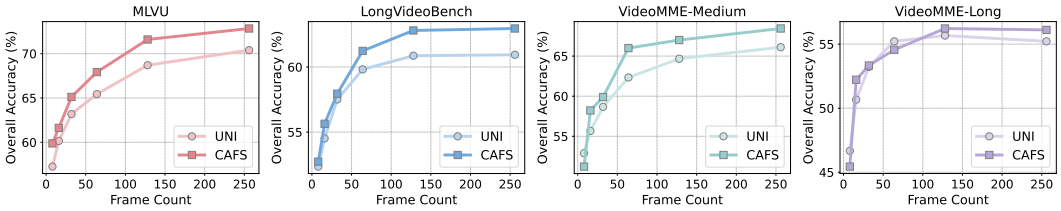

Figure 7: Comparison of CAFS and uniform sampling (UNI) in **DIG** with Qwen2.5-VL-7B [3].

Let f_j denote the frame indexed by j, and let V_j represent its feature vector obtained via DINOv2 [29]. We define the set of *r-frames* as $\{f_{I_i}\}_{i=1}^{N}$ with indices $\{I_i\}_{i=1}^{N}$. To assess their effectiveness in capturing the high-level semantic content within a video, we introduce two quantitative metrics.

Localized Coverage (LoC). This metric assesses the effectiveness with which each *r-frame* captures its local temporal visual context. More specifically, for each *r-frame* f_{I_i}, four neighboring frames are sampled uniformly from its surrounding temporal window. The LoC score is then computed as the average similarity between the *r-frame* and its sampled neighbors across all *r-frames*.

$$\text{LoC} = \frac{1}{4N} \sum_{i=1}^{N} \sum_{j=0}^{3} \text{sim}\left(V_{I_i}, V_{M_{i,j}}\right), \quad \text{where } M_{i,j} = I_i + \left(j - \frac{3}{2}\right) \cdot \left\lfloor \frac{I_{i+1} - I_{i-1}}{6} \right\rfloor \tag{4}$$

Global Coverage (GlC). This metric evaluates how well the *r-frames* collectively represent the entire video content. Ideally, each frame in the video should be similar to at least one *r-frame*. To compute it, we randomly sample 200 frames from the video, denoted as $\{f_x\}_{x \in \mathcal{X}}$. For each frame f_x, we find the maximum similarity to any *r-frame* and average these values across all sampled frames:

$$\text{GlC} = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \max_{i \in [1,N]} \text{sim}(V_{I_i}, V_x) \tag{5}$$

Baseline Selection. We compared two widely adopted baseline methods with CAFS, evaluating their performance using the above metrics across all videos from MLVU [65] and VideoMME [15]. For fair comparison, we ensure all methods select a similar average number of frames per dataset.

- *Uniform Sampling (UNI).* It serves as a common initial reference in various approaches for video understanding tasks, including agent-based systems [46, 47, 54, 14] and temporal search [55].
- *Frame-Per-Second Sampling (FPS).* This method samples a fixed number of frames per second, resulting in a linear increase in total sampled frames with as video length increases.

Analysis. As shown in Figure 6, the performance of uniform sampling declines with increasing video duration. This limitation arises from using a fixed number of frames across videos of varying lengths, which leads to redundancy in short videos and inadequate semantic coverage in long videos. Moreover, while fps sampling maintains stable performance, CAFS consistently outperforms it, particularly for videos over 10 minutes. This indicates that semantic information in videos doesn't grow linearly with length, and that CAFS is more effective at selecting informative frames.

Comparison with Uniform Sampling in DIG. We compare CAFS with uniform sampling within the **DIG** pipeline by replacing CAFS-extracted *r-frames* with uniformly sampled ones. As shown in Figure 7, CAFS consistently outperforms uniform sampling across all benchmarks. In addition,

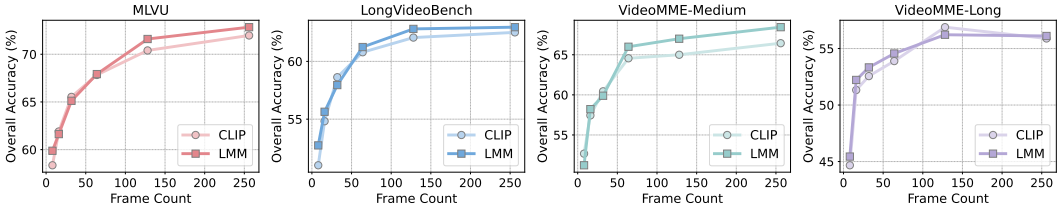

Figure 8: Comparison of LMM reward and CLIPScore [16] in **DIG**. The LMM used for assigning reward and performing final inference is Qwen2.5-VL-7B [3].

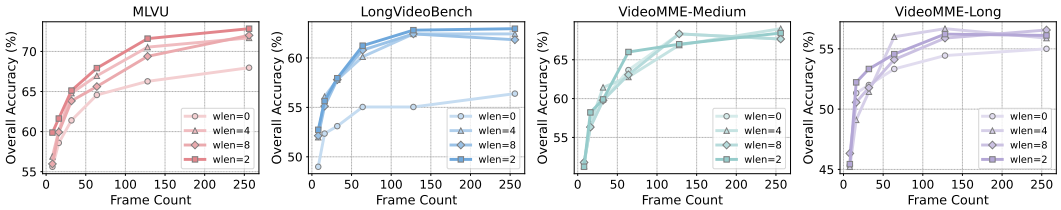

Figure 9: Comparion of different window lengths ($wlen$) in **DIG** with Qwen2.5-VL-7B [3].

The performance gap widens with more input frames, further highlighting the limitation of uniform sampling: for long videos it can't sample sufficient frames to well cover information for subsequent process, while CAFS can adapt for any length of videos and ensures better coverage of video content.

5.5 Ablation Study: Reward Assignment

We evaluate the reward assignment mechanism employed by the LMM in **DIG** by comparing it to a common alternative: computing frame-query similarity using CLIP [32]. Specifically, we substitute all reward values originally assigned by the LMM to the *r-frames* with corresponding CLIPScore [16].

Comparison on LMM and CLIP. As illustrated in Figure 8, the rewards generated by the LMM demonstrate superior performance across all benchmarks. This underscores the LMM's capacity to deliver more precise and semantically rich reward signals through its advanced reasoning abilities. In contrast, CLIPScore [16] depends on superficial feature matching and often fails to capture nuanced or visually complex query requirements.

5.6 Ablation Study: Video Refinement

We conduct an ablation study on our proposed video refinement method, focusing on how different values of $wlen$ affect performance. Specifically, we evaluate the algorithm using various $wlen$ settings and analyze the resulting performance.

Comparison with Different Window Length. As shown in Figure 9, setting $wlen = 0$ results in the worst performance across all benchmarks. This indicates that some queries cannot be effectively addressed using only a single scene, but instead require information from the temporal context. While increasing $wlen$ does not generally lead to consistently improved performance, the setting $wlen = 2$ achieves relatively the best results, particularly on MLVU [65] and LongVideoBench [49]. This suggests that incorporating an appropriate amount of temporal context can be beneficial, while including too much may introduce noise and ultimately degrade model performance.

6 Conclusion

In this work, we first demonstrate that uniform sampling can degrade performance on localized queries. To address this, we introduce **DIG**, a training-free frame selection framework for LMMs that applies uniform sampling across the entire video for global queries, while from a refined subset for localized queries. This subset is constructed via three steps: content-adaptive frame selection (CAFS), reward assignment by the LMM, and video refinement. Extensive experiments across multiple long-form video understanding benchmarks and diverse LMMs show that **DIG** consistently outperforms uniform sampling. However, because **DIG**'s gains depend heavily on the LMM's inherent capabilities, its benefits may be marginal when applied to weaker models.

References

[1] Gpt-4o. https://openai.com/index/hello-gpt-4o/.

[2] K. Ataallah, X. Shen, E. Abdelrahman, E. Sleiman, M. Zhuge, J. Ding, D. Zhu, J. Schmidhuber, and M. Elhoseiny. Goldfish: Vision-language understanding of arbitrarily long videos. In *ECCV*, pages 251–267. Springer, 2024.

[3] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, H. Zhong, Y. Zhu, M. Yang, Z. Li, J. Wan, P. Wang, W. Ding, Z. Fu, Y. Xu, J. Ye, X. Zhang, T. Xie, Z. Cheng, H. Zhang, Z. Yang, H. Xu, and J. Lin. Qwen2.5-vl technical report, 2025.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners, 2020.

[5] W. Chai, E. Song, Y. Du, C. Meng, V. Madhavan, O. Bar-Tal, J.-N. Hwang, S. Xie, and C. D. Manning. Auroracap: Efficient, performant video detailed captioning and a new benchmark, 2025.

[6] G. Chen, Y. Liu, Y. Huang, Y. He, B. Pei, J. Xu, Y. Wang, T. Lu, and L. Wang. Cg-bench: Clue-grounded question answering benchmark for long video understanding, 2024.

[7] L. Chen, X. Wei, J. Li, X. Dong, P. Zhang, Y. Zang, Z. Chen, H. Duan, Z. Tang, L. Yuan, et al. Sharegpt4video: Improving video understanding and generation with better captions. In *NeurIPS*, volume 37, pages 19472–19495, 2024.

[8] Y. Chen, F. Xue, D. Li, Q. Hu, L. Zhu, X. Li, Y. Fang, H. Tang, S. Yang, Z. Liu, E. He, H. Yin, P. Molchanov, J. Kautz, L. Fan, Y. Zhu, Y. Lu, and S. Han. Longvila: Scaling long-context visual language models for long videos, 2024.

[9] Z. Chen, J. Wu, W. Wang, W. Su, G. Chen, S. Xing, M. Zhong, Q. Zhang, X. Zhu, L. Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, pages 24185–24198, 2024.

[10] Z. Cheng, S. Leng, H. Zhang, Y. Xin, X. Li, G. Chen, Y. Zhu, W. Zhang, Z. Luo, D. Zhao, and L. Bing. VideoLLaMA 2: Advancing Spatial-Temporal Modeling and Audio Understanding in Video-LLMs, Oct. 2024.

[11] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel. Palm: Scaling language modeling with pathways, 2022.

[12] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. Scaling instruction-finetuned language models, 2022.

[13] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, et al. The llama 3 herd of models, Aug. 2024.

[14] Y. Fan, X. Ma, R. Wu, Y. Du, J. Li, Z. Gao, and Q. Li. Videoagent: A memory-augmented multimodal agent for video understanding, 2024.

[15] C. Fu, Y. Dai, Y. Luo, L. Li, S. Ren, R. Zhang, Z. Wang, C. Zhou, Y. Shen, M. Zhang, P. Chen, Y. Li, S. Lin, S. Zhao, K. Li, T. Xu, X. Zheng, E. Chen, R. Ji, and X. Sun. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis, 2024.

[16] J. Hessel, A. Holtzman, M. Forbes, R. L. Bras, and Y. Choi. Clipscore: A reference-free evaluation metric for image captioning, 2022.

[17] P. Jin, R. Takanobu, W. Zhang, X. Cao, and L. Yuan. Chat-univi: Unified visual representation empowers large language models with image and video understanding. In *CVPR*, pages 13700–13710, 2024.

[18] W. Kim, C. Choi, W. Lee, and W. Rhee. An image grid can be worth a video: Zero-shot video question answering using a vlm. *IEEE Access*, 2024.

[19] B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, and C. Li. Llava-onevision: Easy visual task transfer, 2024.

[20] X. Li, Y. Wang, J. Yu, X. Zeng, Y. Zhu, H. Huang, J. Gao, K. Li, Y. He, C. Wang, Y. Qiao, Y. Wang, and L. Wang. Videochat-flash: Hierarchical compression for long-context video modeling, 2025.

[21] B. Lin, Y. Ye, B. Zhu, J. Cui, M. Ning, P. Jin, and L. Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023.

[22] K. Lin, F. Ahmed, L. Li, C.-C. Lin, E. Azarnasab, Z. Yang, J. Wang, L. Liang, Z. Liu, Y. Lu, et al. Mm-vid: Advancing video understanding with gpt-4v (ision). *arXiv preprint arXiv:2310.19773*, 2023.

[23] H. Liu, C. Li, Q. Wu, and Y. J. Lee. Visual instruction tuning, 2023.

[24] J. Liu, Y. Wang, H. Ma, X. Wu, X. Ma, X. Wei, J. Jiao, E. Wu, and J. Hu. Kangaroo: A powerful video-language model supporting long-context video input, 2024.

[25] S. Liu, C. Zhao, T. Xu, and B. Ghanem. Bolt: Boost large vision-language model without training for long-form video understanding, 2025.

[26] M. Maaz, H. Rasheed, S. Khan, and F. S. Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023.

[27] J. Min, S. Buch, A. Nagrani, M. Cho, and C. Schmid. Morevqa: Exploring modular reasoning models for video question answering. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13235–13245, June 2024.

[28] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, et al. Gpt-4 technical report, Mar. 2024.

[29] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[30] B. Peng, C. Li, P. He, M. Galley, and J. Gao. Instruction tuning with gpt-4, 2023.

[31] T. Qu, L. Tang, B. Peng, S. Yang, B. Yu, and J. Jia. Does your vision-language model get lost in the long video sampling dilemma?, 2025.

[32] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision, 2021.

[33] S. Ren, L. Yao, S. Li, X. Sun, and L. Hou. 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*, pages 14313–14323, 2024.

[34] X. Shen, Y. Xiong, C. Zhao, L. Wu, J. Chen, C. Zhu, Z. Liu, F. Xiao, B. Varadarajan, F. Bordes, Z. Liu, H. Xu, H. J. Kim, B. Soran, R. Krishnamoorthi, M. Elhoseiny, and V. Chandra. Longvu: Spatiotemporal adaptive compression for long video-language understanding, Oct. 2024.

[35] X. Shen, Y. Xiong, C. Zhao, L. Wu, J. Chen, C. Zhu, Z. Liu, F. Xiao, B. Varadarajan, F. Bordes, Z. Liu, H. Xu, H. J. Kim, B. Soran, R. Krishnamoorthi, M. Elhoseiny, and V. Chandra. Longvu: Spatiotemporal adaptive compression for long video-language understanding, 2024.

[36] M. Shi, F. Liu, S. Wang, S. Liao, S. Radhakrishnan, Y. Zhao, D.-A. Huang, H. Yin, K. Sapra, Y. Yacoob, et al. Eagle: Exploring the design space for multimodal llms with mixture of encoders. *arXiv preprint arXiv:2408.15998*, 2024.

[37] M. Shi, S. Wang, C.-Y. Chen, J. Jain, K. Wang, J. Xiong, G. Liu, Z. Yu, and H. Shi. Slow-fast architecture for video multi-modal large language models, 2025.

[38] E. Song, W. Chai, G. Wang, Y. Zhang, H. Zhou, F. Wu, H. Chi, X. Guo, T. Ye, Y. Zhang, Y. Lu, J.-N. Hwang, and G. Wang. Moviechat: From dense token to sparse memory for long video understanding, 2024.

[39] E. Song, W. Chai, G. Wang, Y. Zhang, H. Zhou, F. Wu, H. Chi, X. Guo, T. Ye, Y. Zhang, Y. Lu, J.-N. Hwang, and G. Wang. Moviechat: From dense token to sparse memory for long video understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18221–18232, Seattle, WA, USA, June 2024. IEEE.

[40] H. Sun, S. Lu, H. Wang, Q.-G. Chen, Z. Xu, W. Luo, K. Zhang, and M. Li. Mdp3: A training-free approach for list-wise frame selection in video-llms, 2025.

[41] X. Tang, J. Qiu, L. Xie, Y. Tian, J. Jiao, and Q. Ye. Adaptive keyframe sampling for long video understanding, 2025.

[42] S. Tong, E. Brown, P. Wu, S. Woo, M. Middepogu, S. C. Akula, J. Yang, S. Yang, A. Iyer, X. Pan, Z. Wang, R. Fergus, Y. LeCun, and S. Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms, 2024.

[43] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models, 2023.

[44] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, Y. Fan, K. Dang, M. Du, X. Ren, R. Men, D. Liu, C. Zhou, J. Zhou, and J. Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, 2024.

[45] X. Wang, Q. Si, J. Wu, S. Zhu, L. Cao, and L. Nie. Retake: Reducing temporal and knowledge redundancy for long video understanding, 2025.

[46] X. Wang, Y. Zhang, O. Zohar, and S. Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent, 2024.

[47] Z. Wang, S. Yu, E. Stengel-Eskin, J. Yoon, F. Cheng, G. Bertasius, and M. Bansal. Videotree: Adaptive tree-based video representation for llm reasoning on long videos, 2025.

[48] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

[49] H. Wu, D. Li, B. Chen, and J. Li. Longvideobench: A benchmark for long-context interleaved video-language understanding, 2024.

[50] H. Wu, H. Liu, Y. Qiao, and X. Sun. Dibs: Enhancing dense video captioning with unlabeled videos via pseudo boundary enrichment and online refinement. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18699–18708, June 2024.

[51] M. Xu, M. Gao, Z. Gan, H.-Y. Chen, Z. Lai, H. Gang, K. Kang, and A. Dehghan. Slowfast-llava: A strong training-free baseline for video large language models, 2024.

[52] S. Yan, T. Zhu, Z. Wang, Y. Cao, M. Zhang, S. Ghosh, Y. Wu, and J. Yu. Videococa: Video-text modeling with zero-shot transfer from contrastive captioners, 2023.

[53] A. Yang, A. Nagrani, P. H. Seo, A. Miech, J. Pont-Tuset, I. Laptev, J. Sivic, and C. Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning, Mar. 2023.

[54] Z. Yang, D. Chen, X. Yu, M. Shen, and C. Gan. Vca: Video curious agent for long video understanding, 2025.

[55] J. Ye, Z. Wang, H. Sun, K. Chandrasegaran, Z. Durante, C. Eyzaguirre, Y. Bisk, J. C. Niebles, E. Adeli, L. Fei-Fei, J. Wu, and M. Li. Re-thinking temporal search for long-form video understanding, 2025.

[56] S. Yu, C. Jin, H. Wang, Z. Chen, S. Jin, Z. Zuo, X. Xu, Z. Sun, B. Zhang, J. Wu, H. Zhang, and Q. Sun. Frame-voyager: Learning to query frames for video large language models, Oct. 2024.

[57] B. Zhang, K. Li, Z. Cheng, Z. Hu, Y. Yuan, G. Chen, S. Leng, Y. Jiang, H. Zhang, X. Li, P. Jin, W. Zhang, F. Wang, L. Bing, and D. Zhao. Videollama 3: Frontier multimodal foundation models for image and video understanding, 2025.

[58] C. Zhang, T. Lu, M. M. Islam, Z. Wang, S. Yu, M. Bansal, and G. Bertasius. A simple llm framework for long-range video question-answering. *arXiv preprint arXiv:2312.17235*, 2023.

[59] H. Zhang, X. Li, and L. Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding, 2023.

[60] P. Zhang, K. Zhang, B. Li, G. Zeng, J. Yang, Y. Zhang, Z. Wang, H. Tan, C. Li, and Z. Liu. Long context transfer from language to vision, 2024.

[61] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.

[62] Y. Zhang, J. Wu, W. Li, B. Li, Z. Ma, Z. Liu, and C. Li. Video instruction tuning with synthetic data, 2024.

[63] Z. Zhao, H. Lu, Y. Huo, Y. Du, T. Yue, L. Guo, B. Wang, W. Chen, and J. Liu. Needle in a video haystack: A scalable synthetic evaluator for video mllms, 2025.

[64] Y. Zhong, J. Xiao, W. Ji, Y. Li, W. Deng, and T.-S. Chua. Video question answering: Datasets, algorithms and challenges, 2022.

[65] J. Zhou, Y. Shu, B. Zhao, B. Wu, Z. Liang, S. Xiao, M. Qin, X. Yang, Y. Xiong, B. Zhang, T. Huang, and Z. Liu. Mlvu: Benchmarking multi-task long video understanding, 2025.

[66] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models, Oct. 2023.

[67] J. Zhu, W. Wang, Z. Chen, Z. Liu, S. Ye, L. Gu, H. Tian, Y. Duan, W. Su, J. Shao, Z. Gao, E. Cui, X. Wang, Y. Cao, Y. Liu, X. Wei, H. Zhang, H. Wang, W. Xu, H. Li, J. Wang, N. Deng, S. Li, Y. He, T. Jiang, J. Luo, Y. Wang, C. He, B. Shi, X. Zhang, W. Shao, J. He, Y. Xiong, W. Qu, P. Sun, P. Jiao, H. Lv, L. Wu, K. Zhang, H. Deng, J. Ge, K. Chen, L. Wang, M. Dou, L. Lu, X. Zhu, T. Lu, D. Lin, Y. Qiao, J. Dai, and W. Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models, 2025.

[68] O. Zohar, X. Wang, Y. Dubois, N. Mehta, T. Xiao, P. Hansen-Estruch, L. Yu, X. Wang, F. Juefei-Xu, N. Zhang, S. Yeung-Levy, and X. Xia. Apollo: An exploration of video understanding in large multimodal models, 2024.

Supplementary Material

The supplementary material is structured as follows:

- The benchmark details in Section A.
- The detailed query classification process in Section B.
- The prompt design of **DIG** in Section C.
- The details of ablation study on query identification in Section D.
- Further details on the ablation study of CAFS in Section E.

A Benchmark Details

In this section, we provide a comprehensive overview of the benchmarks used in our study. The data statistics of each benchmark are summarized in Table 1.

MLVU. MLVU [65] is a multi-task benchmark consisting of 3,102 questions across 9 categories, specifically designed for long video understanding. The dataset is split into two parts: a dev set with 2,593 questions and a test set with 509 questions. The tasks in MLVU are grouped into three main types: 1) holistic tasks, which require analysis of the entire video to derive an answer; 2) single-detail tasks, which hinge on identifying and interpreting one key moment in the video; 3) multi-detail tasks, which demand the integration and analysis of multiple significant segments throughout the video. We use only the multiple-choice questions from the dev set and exclude the open-ended questions.

LongVideoBench. LongVideoBench [49] is a question-answering benchmark that includes 3,763 web-collected videos of varying lengths covering diverse topics, along with 6,678 human-annotated multiple-choice questions spanning 17 fine-grained categories. It is designed to comprehensively evaluate the long-form video understanding capabilities of LMMs. The main challenge involves accurately retrieving and reasoning over detailed multimodal information from lengthy inputs, a task referred to as referring reasoning. In our study, we utilize only the validation set of LongVideoBench [49].

VideoMME. VideoMME [15] is a comprehensive multi-modal evaluation benchmark for LMMs in video analysis. It covers 6 primary visual domains, each with 5 subfields, totaling 30 subdomains. The benchmark includes videos of varying durations—short, medium, and long-term—ranging from 11 seconds to 1 hour. In addition to video frames, it incorporates other modalities such as subtitles and audio. The dataset comprises 900 videos, amounting to approximately 254 hours of content, and features a total of 2,700 question-answer pairs. In our study, we use only the video and the corresponding questions, without leveraging any additional modalities such as subtitles. Unless otherwise specified, we evaluate across all benchmarks spanning the three duration splits: short, medium, and long.

Table 1: **Dataset Statistics.** Overview of the data statistics across benchmarks: LongVideoBench, MLVU and VideoMME. For LongVideoBench, the statistics correspond to the validation set.

Dataset	Avg. Duration (s)	Data Size
MLVU [65]	636.2	2174
LongVideoBench-val [49]	732.2	1337
VideoMME-short [15]	80.7	900
VideoMME-medium [15]	516.8	900
VideoMME-long [15]	2466.3	900

B Query Classification

In this section, we elaborate on the query classification process described in Section 3.

MLVU. For MLVU [65], since its design closely aligns with our definition as discussed in Section A, we classify the queries as follows: queries that in holistic tasks are considered global queries, while those that in single-detail or multi-detail tasks are classified as localized queries. Following this approach, we identify a total of 462 global queries and 1708 localized queries.

Query Identification Prompt

You are a helpful assistant in a video-based question-answering process. Your task is to assess whether the given query requires a comprehensive understanding of the entire video or can be effectively addressed by focusing on specific video segments, based on the contextual cues embedded within the query.
Query: <Q>
In your response, first provide a brief analysis based on the query, and then make your judgement.

Figure 10: **Query Identification Prompt.** The LMM is first provided with the task definition, followed by an application of the chain-of-thought [48] technique to arrive at a judgment.

Reward Assignment Prompt

You are acting as a reward model designed to guide the video-based question-answering process. The video has a duration of <T> seconds, and your input is a sampled frame taken at the timestamp <I> seconds.
Query: <Q>; Frame <F>
Please perform the following steps to finish your evaluation:
1. Describe the visual content of the sampled frame, focusing on elements relevant to the query, if such elements are present.
2. Assign a relevance reward between 0 and 100 based on: (1) The sampled frame's direct usefulness in answering the query (2) Whether the frame suggests that adjacent frames might provide additional information that helps answer the query more effectively.

Figure 11: **Reward Assignment Prompt.** The LMM is first presented with the task definition and associated metadata. Then, the chain-of-thought reasoning technique [48] is applied to assign the reward for the input frame.

LongVideoBench. Due to the design of LongVideoBench[49], which focuses on referring reasoning and evaluates a model's ability to reason over detailed visual information, the query format in this benchmark aligns with our definition of localized queries. Therefore, all queries in LongVideoBench [49] are considered as localized queries.

VideoMME. In VideoMME [15], there are no existing definitions for localized or global queries. Therefore, we manually annotate the queries based on the following instruction: A query is classified as localized if it contains specific references to objects (e.g., "What is the man holding an umbrella doing?"), scenes (e.g., "What is the woman doing at the table?"), or abstract concepts tied to specific elements in the video (e.g., "What magic trick is being performed in the video?"). Queries that do not focus on specific details and instead seek general information about the video as a whole are considered global. Through this annotation process, we classify 479 global queries and 2221 localized queries.

C Prompt Design

In this section, we present the prompt design of our **DIG** framework. Figure 10 illustrates the prompt used by LMMs to determine whether a given query is global or localized in nature. Additionally, Figure 11 displays the prompt employed by LMMs for assigning a reward score to each input frame. Furthermore, the general prompt template used across all experiments in our study for enabling LMMs to perform direct inference is illustrated in Figure 12.

As shown in Figure 10, the LMM is first presented with the task definition. It is then asked to determine whether the given query is global or localized. Importantly, we employ the chain-of-thought prompting strategy [48], which guides the model to first analyze the query in detail, explicitly articulate its reasoning process, and finally provide a well-justified judgment.

Inference Prompt

What is the video mainly about?
A. Planes invented by the Wright Brothers.
B. The structural difference between the planes created by Whitehead and planes created by the Wright Brothers.
C. Who invented the first plane.
D. How Whitehead and the Wright Brothers cooperated to invent the first motorized flight.
Please select the best answer from the options provided and directly provide the letter representing your choice without giving any explanation.

Figure 12: **Prompt Template Example.** Example of the prompt template used by LMMs to perform direct inference.

As illustrated in Figure 11, the LMM is first provided with the task definition along with information about the frame, including the video duration and the frame's timestamp within the video. Given the query and the corresponding visual frame, the model is instructed to first describe the visual content of the frame, focusing on elements relevant to the query. Following this analysis, it assigns a reward score based on two detailed evaluation criteria outlined in Figure 11.

D More Details in Ablation Study: Query Identification

In this section, we present experimental results evaluating the ability of LMMs to determine whether a given query is global or localized. Specifically, we assess how well these models align with human annotators by computing their accuracy on different query types from MLVU [65], LongVideoBench [49], and VideoMME [15]. These query types were classified by human annotators, as detailed in Section B. We evaluate two versions of Qwen-VL: Qwen2.5-VL-7B [3] and Qwen2-VL-7B [44]. To investigate how model scaling affects alignment with human judgment, we also include results for GPT-4o-mini [1].

Localized Queries Are Easier to Identify. As shown in Table 2, all evaluated LMMs achieve higher accuracy on localized queries than on global queries. This suggests that current LMMs are better aligned with human judgment when identifying queries that focus on specific video segments or timepoints. In contrast, they struggle with recognizing global queries, which typically require holistic understanding of the entire video content.

LMM Advancements Improve LQ Identification More. Interestingly, as the performance of LMMs improves, the identification accuracy for LQs increases correspondingly. However, this trend does not hold for GQs, where improvements in model capabilities do not consistently translate to better alignment with human annotations. This indicates that identifying global queries remains a challenging task even for advanced LMMs.

Table 2: Accuracy (%) of different LMMs in identifying localized queries (LQ) and global queries (GQ) across multiple benchmarks.

LMM	MLVU [65]		LongVideoBench [49]		VideoMME [15]	
	LQ	GQ	LQ	GQ	LQ	GQ
GPT-4o-mini [3]	88.14	38.10	91.47	/	93.56	78.08
Qwen2.5-VL-7B [3]	79.95	51.95	95.43	/	87.80	83.92
Qwen2-VL-7B [3]	68.79	52.81	73.82	/	80.64	62.42

E More Details in Ablation Study: CAFS

In this section, we provide additional insights into the ablation study for CAFS by reporting its compression performance across multiple video benchmarks. Specifically, for a given video containing

N total frames, let T denote the number of frames selected by CAFS. We define the *compression rate* as the ratio of the number of selected frames to the total number of frames:

$$\text{Compression Rate} = \frac{T}{N}. \tag{6}$$

As shown in Table 3, the compression rates achieved by CAFS across three benchmark are all around 1%. This indicates that CAFS is able to extract semantic information and capture major visual content using only a small fraction of the original frames. This high compression rate also highlights the significant amount of temporal redundancy present in most videos. By intelligently selecting informative frames, CAFS effectively reduces redundant information while preserving the essential visual and semantic content of the video.

Table 3: Compression rates (%) of CAFS on different video benchmarks.

Method	MLVU [65]	LongVideoBench [49]	VideoMME [15]
CAFS	0.870	0.639	0.795