EchoPrune: Interpreting Redundancy as Temporal Echoes for Efficient VideoLLMs

Jiameng Li[1] **Minye Wu**[1] **Jiezhang Cao**[2*] **Aleksei Tiulpin**[3] **Matthew B. Blaschko**[1]
[1]KU Leuven, [2]Shanghai Jiao Tong University, [3]Weill Cornell Medicine
https://github.com/renaissanceee/EchoPrune

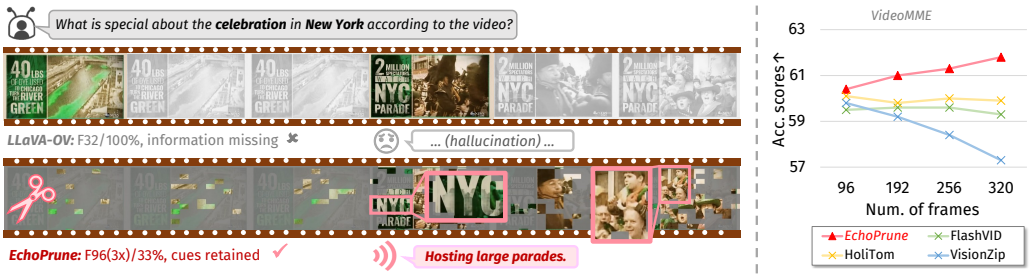

Figure 1: **Left:** EchoPrune extends the temporal resolution with more visible frames for fine-grained video understanding, then selects pivot tokens to fit the budgets (F32 is short for 32 frames). **Right:** Under the default token footprint, EchoPrune scales visible frames up to boost SOTA performance.

Abstract

Long-form video understanding remains challenging for Video Large Language Models (VideoLLMs), as the dense frame sampling introduces massive visual tokens while sparse sampling risks missing critical temporal evidence and leading to LLM hallucination. Existing training-free token reduction methods either treat videos equally as static images or rely on segment-level merging heuristics, which weaken fine-grained spatiotemporal modeling and introduce additional overhead. In this paper, we propose ***EchoPrune***, a lightweight and training-free token pruning method that improves temporal resolution under a fixed LLM-side visual token budget. Our core idea is to interpret redundant video tokens as temporal echoes: if a token is well reconstructed from the previous frame, it is merely a temporally redundant echo; otherwise, it may capture new events, motion, or query-relevant visual evidence. Based on this insight, EchoPrune scores visual tokens by *(i)* query-guided crossmodal relevance and *(ii)* temporal reconstruction error, measured by correspondence matching and echo matching across consecutive frames. The selected tokens preserve task-relevant cues and temporal novelty while suppressing predictable redundancy, allowing VideoLLMs to observe more frames without increasing the decoding budget. Extensive experiments on LLaVA-OV, Qwen2.5VL, and Qwen3VL across six video understanding benchmarks show that EchoPrune enables VideoLLMs to process up to $\mathbf{20}\times$ frames under the same token budget, yielding improved performance ($\mathbf{5}\%^{\uparrow}$) and inference speedup ($\mathbf{5.6}\times$ for prefilling) on LLaVA-OV-7B.

1 Introduction

Video understanding has increasingly benefited from the integration of large language models, equipping Video Large Language Models (VideoLLMs) with strong multimodal reasoning capabilities

*Correspondence

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

[18, 48]. In VideoLLMs, a vision encoder [63] transforms sampled video frames into token sequences for LLM decoding. However, such a dense tokenization introduces substantial redundancy, as consecutive frames exhibit high temporal correlation. The resulting token proliferation significantly increases computational costs and limits practical deployments [13]. Therefore, eliminating redundant video tokens while preserving task-relevant semantic information is critical for improving both the efficiency and applicability of VideoLLMs [5, 39].

Existing visual token reduction methods [15, 40, 56] can be categorized into generic branches that treat images and videos equally, and video-specific ones considering the inductive bias in video data. Generic methods select tokens based on attention salience [58] or subset coverage [19], while video-specific methods exploit temporal structures via segment-level pruning [40], token merging [39], uniqueness estimation [62], or spatiotemporal organization [5]. Despite their progress, these methods often rely on indirect or fixed criteria, such as miscalibrated attention [55, 61], predefined partitions or global pooling [28, 39, 40], and non-adaptive uniqueness measures [62]. Such designs can disrupt temporal continuity, weaken inter-segment information exchange, and discard fine-grained temporal cues that are essential for video understanding. As a result, they may suffer from temporal information loss and performance degradation, especially on long videos where relevant evidence is sparsely distributed over time [20]. Moreover, some of these methods [40] adopt DPC-KNN clustering [37] or two-stage pipelines [5] from projection-layer pruning followed by inner-LLM merging, which introduces non-negligible computational overhead with implementation difficulty.

The central challenge is to mitigate the information bottleneck when representing abundant but unevenly distributed video content with compact visual tokens [5, 39, 62]. Sparse frame sampling is commonly adopted to avoid prohibitive token growth [18]. However, it may discard critical visual evidence and introduce a semantic gap, causing a blind LLM to rely on language priors and hallucinate. This is illustrated in Fig. 1 (left), where F32 means uniformly sampling 32 frames from a video clip along the temporal axis. This suggests that video pruning should not be treated merely as a means of reducing computation. Instead, the designed pruning is supposed to improve the temporal resolution [5, 16], allowing the model to observe more frames under the same LLM-side token budget. Meanwhile, token importance in multimodal video understanding is inherently query-dependent [4, 43]. Therefore, an effective pruning strategy ought to preserve query-relevant evidence, capture newly emerging visual information, and suppress repetitive or predictable content with minimal compression overhead.

In this paper, we propose ***EchoPrune***, a query-aware and lightweight video token pruning method that preserves informative visual evidence while suppressing temporal redundancy. Our key insight is that redundant video tokens can be interpreted as *temporal echoes*: if a token in the current frame can be well predicted by tokens from the preceding frame, it is likely to carry repetitive information; otherwise, it may correspond to newly emerging events, local motion, or task-relevant visual evidence. To operationalize this idea, we formulate video token pruning under the framework of *Maximum Marginal Relevance (MMR)* [3], where the importance of each token is determined by both its relevance to the user query and its redundancy regarding preceding visual context. Specifically, EchoPrune scores each video token from three complementary perspectives. First, *query-guided crossmodal relevance* measures the semantic alignment between visual tokens and the textual query, encouraging the model to preserve visual evidence needed for answering the question. Second, *correspondence matching* captures local content changes by comparing tokens across adjacent frames, which helps identify motion-related visual information. Third, *echo matching* reconstructs the current token from a spatial neighborhood in the preceding frame, providing a lightweight proxy for temporal predictability. These two types of matching form the redundancy proxy called *temporal reconstruction error*. By integrating these scores, EchoPrune performs once-for-all Top-K pruning before the LLM, without additional training or intrusive modifications to the pretrained VideoLLM. As shown in Fig. 1 (right), under the same token budget, EchoPrune continues benefiting from using more frames, whereas competing methods tend to saturate or degrade, indicating richer temporal evidence preservation for stronger long video understanding.

Our key contributions are summarized as follows:

- We propose ***EchoPrune***, a training-free and lightweight video token pruning method that preserves task-relevant visual evidence while suppressing temporal redundancy.
- We interpret video redundancy as *temporal echoes* and combine query-guided relevance, correspondence matching, and echo matching for a holistic Top-K token selection (Sec. 3).

- Extensive experiments on LLaVA-OV, Qwen2.5VL, and Qwen3VL show that ***EchoPrune*** supports up to $20\times$ more frames under the same token budget, achieves SOTA performance, and accelerates inference with up to $5.6\times$ prefilling and $2.1\times$ TTFT speedup on LLaVA-OV (Sec. 4).

2 Preliminaries

Pruning stages. Grounded in the theory of information bottleneck [49], many pruning methods intervene in the projection layer [40, 62], where multimodal tokens are aligned in a shared embedding space but have not yet fused through cross-attention. Meanwhile, empirical evidence from FrameFusion [7] suggests that the ranking of token similarity hardly changes across LLM layers. While HoliTom [39] and FlashVID [5] introduce inner-LLM merging to further exploit cross-modal interactions in LLMs, this two-stage pipeline inevitably increases computational overhead and implementation complexity. To achieve *once-for-all* effects, our pruning is conducted once in projection space like FastVID and UniComp [40, 62].

Maximum marginal relevance (MMR). Video token pruning can be viewed as a subset selection problem, aiming to find a representative subset maximizing information coverage. Specifically, the pruning objective can be characterized as a submodular function which satisfies the *diminishing returns* property [8], *i.e.*, the incremental benefit of incorporating a new video token decreases as the selected set grows. This mathematical foundation motivates our adoption of the MMR framework [3], whose scoring function incorporates both relevance terms for task-oriented importance and redundancy terms to capture the diminishing gain. Among video pruning methods, MMG-Vid [31] calculates coarse frame-level MMR for frame budgets. SemVID [20] derives MMR scores as the query evidence and intra-frame redundancy, while its greedy search relies on predefined segments and an anchor set. By contrast, our EchoPrune formulates the redundancy term as the reconstruction error from the history frame and conducts a single holistic ranking, which yields better efficiency.

Frame matching and reconstruction. As a cornerstone of computer vision, frame matching has been adapted in token pruning for VideoLLMs. MMG-Vid [31] studies frame-level similarity, and subsequent methods shift towards token-level uniqueness to analyze temporal dependencies across consecutive frames. Denote $\mathbf{v}_i^k$ as the i_{th} token in frame k, SemVID [20] and KiToke [11] generally quantify temporal connectivity via the L_2 distance, written as $L_2(\mathbf{v}_i^k, \mathbf{v}_i^{k-1})$. KiToke [11] further evaluates spatial displacement by the minimum L_2 distance between a token and all candidates in the preceding frame, *i.e.*, $\min L_2(\mathbf{v}_i^k, \mathbf{v}_j^{k-1})$. In comparison, we replace the minimal aggregation with echo matching to preserve fine-grained semantics.

3 Methods

Video tokens are selected according to their MMR scores following the procedure in Algorithm 1. As shown in Fig. 2 (left), the narrator in the video is essential to answer the question `"What is the man doing?"` (relevance), while the background lamp persists across frames with less temporal information (redundancy). This observation motivates our MMR-based token selection. In the task of Visual Question Answering (VQA), a straightforward MMR score of token $\mathbf{v}_i$ can be written as:

$$S_i = R(\mathbf{v}_i, \mathcal{T}) - D(\mathbf{v}_i, \mathcal{V}_{\text{S}}). \tag{1}$$

Here, the relevance term (R) aggregates over the text set $\mathcal{T}$, while the redundancy term (D) measures the marginal contribution to the selected vision set $\mathcal{V}_{\text{S}}$. To estimate the **redundancy**, EchoPrune models the current frame as a *temporal echo* of its predecessor. By causally propagating the information from earlier to later frames, we derive the reconstruction error between adjacent frames as a self-supervised metric to filter out predictable (*i.e.*, replaceable) tokens. Our reconstruction is guided by two complementary concerns: *(i) temporal novelty:* tokens in frame k that are poorly reconstructed from the last frame tokens $\mathcal{V}^{k-1}$ indicate the arrival of a new event with semantic importance; *(ii) spatial motion:* a substantial semantic shift for inter-frame tokens at the same position suggests the presence of dynamic objects. Consequently, we calculate reconstruction error through echo matching (δ_{echo}) and correspondence matching (δ_{corr}) to quantify the temporal novelty and spatial motion. The **relevance** term is defined as the maximum aggregation from crossmodal relevance (r). Our pipeline is illustrated in Fig. 2. Let $\mathbf{v}_i^k$ denote a token in frame k at position i.

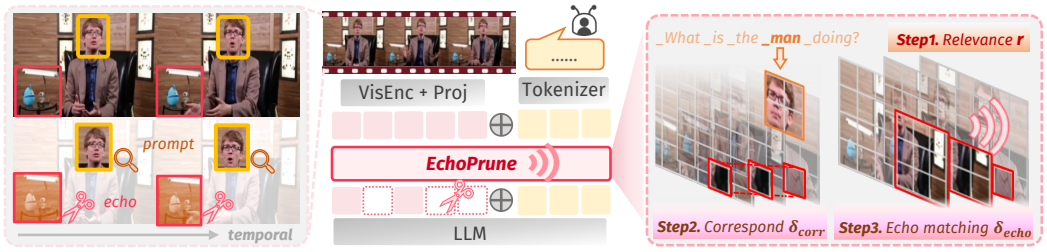

Figure 2: **Overview.** EchoPrune identifies the most informative tokens via a three-fold decomposition: **Step 1.** Crossmodal relevance guided by user query (r); **Step 2.** Spatial motion based on correspondence matching (δ_{corr}); **Step 3.** Temporal novelty driven by echo matching (δ_{echo}).

Following Eqn. (1), EchoPrune frames the MMR score of video tokens $\{\mathbf{v}_i^k\}$ as:

$$R(\mathbf{v}_i^k, \mathcal{T}) \coloneqq r(\mathbf{v}_i^k, \mathcal{T}) \tag{2}$$

$$D(\mathbf{v}_i^k, \mathcal{V}^{k-1}) \coloneqq \delta_{\text{corr}}(\mathbf{v}_i^k, \mathcal{V}^{k-1}) + \delta_{\text{echo}}(\mathbf{v}_i^k, \mathcal{V}^{k-1}). \tag{3}$$

The definitions of these reconstruction errors are presented as follows.

3.1 Crossmodal Relevance

The query-guided crossmodal relevance term is formulated as an attention-like interaction between a normalized visual token $\tilde{\mathbf{v}}_i^k$ and a normalized textual token $\tilde{\mathbf{t}}_j$. As observed in MI-Pruner [19], the average pooling tends to flatten the semantic nuance. To preserve the most salient crossmodal signals, we employ maximum aggregation among text tokens $\tilde{\mathcal{T}}$:

$$r\left(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{T}}\right) = \max_{\tilde{\mathbf{t}}_j \in \tilde{\mathcal{T}}} (\tilde{\mathbf{v}}_i^k)^\top \tilde{\mathbf{t}}_j, \tag{4}$$

where $\tilde{\cdot}, \cdot^\top$ denote L_2 normalization and transposition. The relevance term measures the token's multimodal significance in the VQA task. We show its importance in the ablation study (Sec. 4.3).

3.2 Temporal Reconstruction

We evaluate the video redundancy through two types of reconstruction error, δ_{corr} and δ_{echo}. To establish a temporal correspondence of $\mathbf{v}_i^k$, we first define the *correspondence reconstruction error* δ_{corr} to directly measure the temporal persistence at the same position i:

$$\delta_{\text{corr}}\left(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1}\right) = (\tilde{\mathbf{v}}_i^k)^\top \tilde{\mathbf{v}}_i^{k-1}. \tag{5}$$

The negative sign is deferred to the final score Eqn. (11). Then, we construct a similarity matrix between $\mathbf{v}_i^k$ and last-frame candidates $\{\mathbf{v}_j^{k-1}\}$. Following the energy-based formulation, the temperature scaled similarity score $\boldsymbol{\rho}_{ij}^k$ is written as:

$$\boldsymbol{\rho}_{ij}^k = \frac{(\tilde{\mathbf{v}}_i^k)^\top \tilde{\mathbf{v}}_j^{k-1}}{\tau}, \tag{6}$$

where τ serves as the temperature parameter. To enforce local spatiotemporal coherence and reduce computational overhead, we restrict the candidate set of $\{\mathbf{v}_j^{k-1}\}$ to a spatial neighborhood Ω_i^ϵ centered at position i with window size ϵ, *e.g.*, 3×3. Then we apply softmax over candidates from frame $k-1$ to obtain the conditional probability p_{ij}^k:

$$p_{ij}^k \triangleq p\left(\tilde{\mathbf{v}}_j^{k-1} | \tilde{\mathbf{v}}_i^k\right) = \underset{j \in \Omega_i^\epsilon}{\text{softmax}}\left(\boldsymbol{\rho}_{ij}^k\right), \tag{7}$$

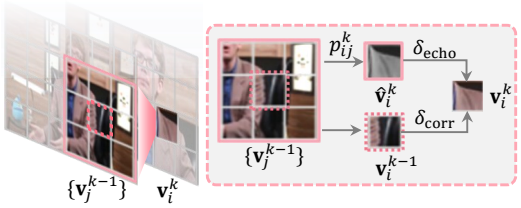

Figure 3: Temporal reconstruction ($\tilde{\cdot}$ omitted).

which serves as the matching weight for the temporal echo. The token $\tilde{\mathbf{v}}_i^k$ can be approximated (denoted as $\hat{\tilde{\mathbf{v}}}_i^k$) by a linear combination of its temporal neighbors in frame $k-1$:

$$\hat{\tilde{\mathbf{v}}}_i^k = \sum_{j \in \Omega_i^\epsilon} p_{ij}^k \tilde{\mathbf{v}}_j^{k-1}. \tag{8}$$

Consistent with Eqn. (5), we define the *echo reconstruction error* δ_{echo} as the inner product between the original token and its reconstructed counterpart:

$$\delta_{\text{echo}}\left(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1}\right) = (\tilde{\mathbf{v}}_i^k)^\top \hat{\tilde{\mathbf{v}}}_i^k = (\tilde{\mathbf{v}}_i^k)^\top \left(\sum_{j \in \Omega_i^\epsilon} p_{ij}^k \tilde{\mathbf{v}}_j^{k-1} \right). \tag{9}$$

As shown in Fig. 3, the overall reconstruction error combines these two components:

$$\delta = \delta_{\text{corr}} + \delta_{\text{echo}}. \tag{10}$$

Algorithm 1 EchoPrune

Input: Dense video tokens $\mathcal{V}=\{\mathbf{v}_i^k\}$, $|\mathcal{V}|=\gamma N_V$; text tokens $\mathcal{T}=\{\mathbf{t}_j\}_{j=1}^{N_T}$; temperature τ; window size ϵ.
Output: Selected video tokens $\mathcal{V}_{\text{S}}$, where $|\mathcal{V}_{\text{S}}|=N_V$.
// Step 1. Crossmodal relevance
$r(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{T}}) \leftarrow \max_{\tilde{\mathbf{t}}_j \in \tilde{\mathcal{T}}} (\tilde{\mathbf{v}}_i^k)^\top \tilde{\mathbf{t}}_j$
// Step 2-3. Intra-modal reconstruction error (redundancy)
$\boldsymbol{\rho}_{ij}^k \leftarrow \tilde{\mathbf{v}}_i^{k\top} \tilde{\mathbf{v}}_j^{k-1}/\tau$ ▷ Scaled similarity
$p_{ij}^k \triangleq p(\tilde{\mathbf{v}}_j^{k-1} | \tilde{\mathbf{v}}_i^k) \leftarrow \text{softmax}_{j \in \Omega_i^\epsilon}(\boldsymbol{\rho}_{ij}^k)$ ▷ Conditional probabilities
$\delta(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1}) \leftarrow (\tilde{\mathbf{v}}_i^k)^\top \tilde{\mathbf{v}}_i^{k-1} + (\tilde{\mathbf{v}}_i^k)^\top (\sum_{j \in \Omega_i^\epsilon} p_{ij}^k \tilde{\mathbf{v}}_j^{k-1})$ ▷ Reconstruction error
// MMR selection
$S(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}, \tilde{\mathcal{T}}) \leftarrow r(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{T}}) - \delta(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1})$
$\mathcal{V}_{\text{S}}=\{\mathbf{v}_i^k\}$, with $|\mathcal{V}_{\text{S}}|=N_V \leftarrow \text{Top-K}_{N_V}(S(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}, \tilde{\mathcal{T}}))$
return $\mathcal{V}_{\text{S}}$.

3.3 Overall MMR Scores

Given Eqn. (4) and Eqn. (10), the MMR score of EchoPrune is written as:

$$S\left(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}, \tilde{\mathcal{T}}\right) = r\left(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{T}}\right) - \delta\left(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1}\right). \tag{11}$$

Under the token budget B, a simple Top-K sorting is sufficient to obtain the selected set $\mathcal{V}_{\text{S}}$.

$$\mathcal{V}_{\text{S}} = \text{Top-K}_B\left(S(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}, \tilde{\mathcal{T}})\right). \tag{12}$$

We denote $N_V=M_F \cdot N_F$ as the default budgets, where N_V consists of M_F frames with N_F tokens/frame, *e.g.*, LLaVA-OV [18] sets $M_F=32$ frames and $N_F=196$ tokens/frame. To improve the temporal resolution under the default budget N_V, we retain $\frac{1}{\gamma}$ tokens from the extended γM_F ($\gamma > 1$) frames, as shown in Algorithm 1. Since the first frame has no reconstruction error, we retain an average of $\frac{N_F}{\gamma}$ tokens based on its crossmodal relevance.

4 Experiments

4.1 Setup

Benchmarks. We evaluate our method on 6 video understanding benchmarks: VideoMME [6] (without subtitle), EgoSchema [34], LongVideoBench [54], MLVU [70], VideoMMMU [10] and LVBench [52]. The evaluation follows the standard settings of LMMs-Eval [66]. We adopt LLaVA-OV [18], instruction-tuned Qwen2.5VL [48] and Qwen3VL [2] as the pretrained VideoLLMs for token pruning. Results on various model scales and settings can be found in App. B.3. To ensure sufficient evidence frames, we increase the temporal resolution with higher frame sampling rates. Specifically, we scale the number of frames up to F320, while keeping the default LLM token budget fixed, *i.e.*, selecting the same amount of tokens from extended frames. This setting avoids undersampling bias and LLM hallucination in F32 benchmarks [39, 40], where evidence frames may already be missed before pruning. See more discussions about temporal sampling in App. A.2.

Comparison methods. We reproduce six comparison methods using their official repositories, which are categorized into: *(i) generic methods:* applicable for both image and video pruning, like attention salience guided VisionZip [58] and subset coverage based MI-Pruner [19]; *(ii) video-specific methods:* considering spatiotemporal processing, *e.g.*, FastVID [40], HoliTom [39], UniComp [62] and FlashVID [5]. Descriptions of the comparison methods can be found in Sec. 5.

Implementation. We test on a single A100 GPU with `batch_size=1` following previous work [5, 62]. For Qwen3VL, we preserve the inter-frame timestamp tokens outside of token budgets. The model architecture affects hyperparameters due to different tokenization strategies in visual encoders. Based on experimental results in the ablation study (Sec. 4.3), we select the optimal parameters for each model family and keep this set of parameters fixed in the comparative experiments.

Table 1: **Performance on LLaVA-OV-7B.** Keeping the F32 budget, F96/33.3% denotes keeping 33.3% tokens from 96 frames. EchoPrune shows SOTA performance on long video benchmarks represented by MLVU and LVBench. Publication venues or preprint dates are indicated in grey.

Method	VideoMME				EgoSchema		LongVideo Bench	MLVU	LVBench	Average	
	Short	Medium	Long	Overall	Subset	Total				Scores	%
Duration	< 3 min	3-30 min	> 30 min	1-60 min	< 3 min	< 3 min	1-60 min	3-120 min	30-120 min	< 120 min	
LLaVA-OV-7B	70.1	56.7	48.9	58.6	62.2	60.3	56.4	47.0	38.4	52.1	100
F96 (3×) / 33.3%											
MI-Pruner [19] 2026.04	70.7	60.7	49.0	60.1	62.8	60.5	55.3	48.0	39.3	52.6	+1.0
VisionZip [58] CVPR'25	71.7	57.9	49.8	59.8	64.2	60.9	57.5	46.8	39.7	52.9	+1.5
FastVID [40] NeurIPS'25	69.4	56.6	48.8	58.3	63.0	59.6	56.3	45.6	40.0	51.9	-0.4
HoliTom [39] NeurIPS'25	**73.0**	58.2	49.1	60.1	**64.8**	61.5	58.3	47.1	40.2	53.4	+2.5
UniComp [62] CVPR'26	71.4	**59.4**	49.6	60.1	63.6	**62.3**	**58.5**	47.5	40.8	**53.8**	**+3.3**
FlashVID [5] ICLR'26	72.4	56.7	49.3	59.5	63.8	61.0	58.4	47.5	41.0	53.5	+2.7
EchoPrune	71.9	58.9	**50.5**	**60.4**	62.3	60.7	58.2	**48.2**	**41.3**	**53.8**	**+3.3**
F256 (8×) / 12.5%											
MI-Pruner [19] 2026.04	66.8	58.1	50.1	58.3	63.0	60.1	53.1	47.4	40.8	51.9	-0.4
VisionZip [58] CVPR'25	68.4	56.8	50.1	58.4	63.5	61.7	54.7	47.0	40.4	52.4	+0.6
FastVID [40] NeurIPS'25	64.6	54.4	48.8	55.9	59.4	57.3	52.3	42.7	39.8	49.6	-3.1
HoliTom [39] NeurIPS'25	**71.8**	57.7	50.4	60.0	64.1	61.2	56.2	47.4	42.9	53.5	+2.7
UniComp [62] CVPR'26	70.7	60.6	50.2	60.5	**64.2**	**62.4**	57.4	49.5	43.1	54.6	+4.8
FlashVID [5] ICLR'26	71.1	57.1	50.7	59.6	63.4	60.1	**58.3**	48.0	42.3	53.7	+3.1
EchoPrune	**71.8**	**61.3**	**50.8**	**61.3**	64.0	61.3	57.8	**49.7**	**43.3**	**54.7**	**+5.0**
F320 (10×) / 10% (ultra)											
MI-Pruner [19] 2026.04	65.8	57.4	49.1	57.4	61.8	59.9	51.0	45.9	40.0	50.8	-2.5
VisionZip [58] CVPR'25	67.0	56.8	48.1	57.3	63.2	60.7	53.4	44.8	40.5	51.3	-1.5
FastVID [40] NeurIPS'25	63.3	53.8	46.9	54.7	56.7	55.9	50.6	41.3	38.7	48.2	-7.5
HoliTom [39] NeurIPS'25	**72.7**	56.8	50.2	59.9	63.8	61.1	56.5	48.3	42.0	53.6	+2.9
UniComp [62] CVPR'26	72.0	59.8	50.3	60.7	**64.4**	**61.2**	**58.5**	47.7	42.0	54.0	+3.6
FlashVID [5] ICLR'26	70.7	56.7	50.4	59.3	63.4	60.8	57.0	47.6	**42.9**	53.5	+2.7
EchoPrune	71.7	**61.1**	**52.6**	**61.8**	63.0	60.4	57.6	**49.8**	**42.9**	**54.5**	**+4.6**

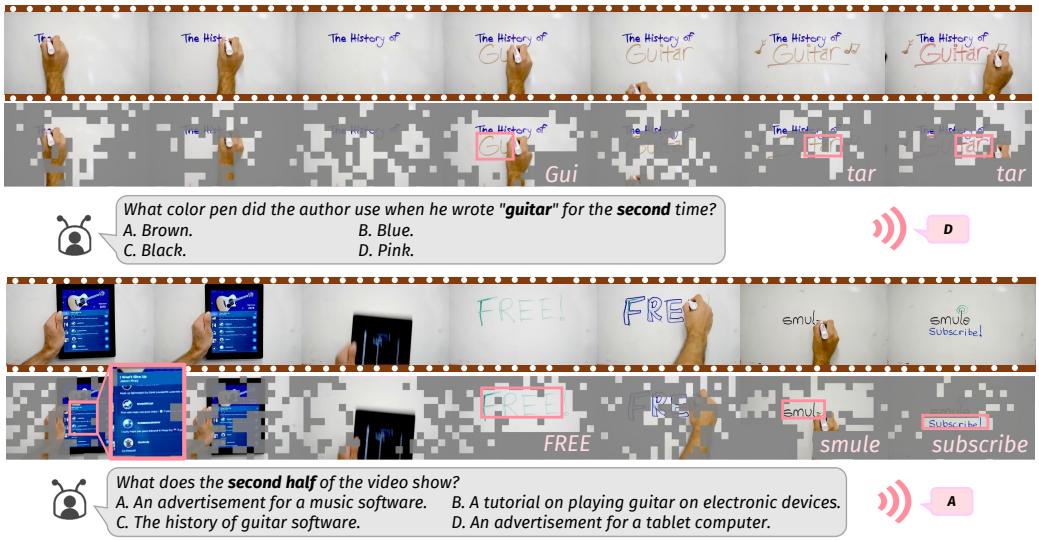

Figure 4: **Pruning effects on Qwen2.5VL-7B**. EchoPrune preserves the count and OCR patch `"-tar"` and identifies the logo `"smule"` and contextual cues `"subscribe"`. Two questions are from the same video in VideoMME (F160/20%). Pivot text cues are highlighted in **bold**.

Table 2: **Performance on Qwen2.5VL-7B.** We keep F16 token budget to compare pruning effects following [5]. EchoPrune demonstrates consistent effectiveness in average scores.

Method	VideoMME				EgoSchema		LongVideo Bench	MLVU	Video MMMU	Average	
Duration	Short < 3 min	Medium 3-30 min	Long > 30 min	Overall 1-60 min	Subset < 3 min	Total < 3 min	1-60 min	3-120 min	< 20 min	Scores < 120 min	%
Qwen2.5VL-7B	66.4	56.4	48.2	57.0	58.2	55.6	56.9	40.6	45.9	51.2	100
F64 (4×) / 25%											
Random	72.0	59.6	51.6	61.0	60.2	57.3	57.6	43.3	44.2	52.7	+2.9
Query	70.4	59.2	**52.4**	60.7	57.4	56.6	54.7	44.4	40.1	51.3	+0.2
MI-Pruner [19] 2026.04	71.8	60.0	51.3	61.0	60.4	57.5	56.5	45.4	43.2	52.7	+2.9
VisionZip [58] CVPR'25	72.3	**60.1**	51.6	61.3	61.0	**58.5**	57.8	44.7	43.1	53.1	+3.7
FastVID [40] NeurIPS'25	71.0	58.3	50.6	60.0	61.4	58.1	57.7	45.0	43.2	53.0	+3.5
FlashVID [5] ICLR'26	**73.0**	59.3	50.3	60.9	60.2	58.4	**58.4**	45.0	43.5	53.2	+3.9
EchoPrune	72.4	**60.1**	51.6	**61.4**	**61.8**	58.1	57.9	**48.3**	**46.8**	**54.5**	**+6.4**
F80 (5×) / 20%											
Random	71.0	61.9	51.0	61.3	59.4	57.4	57.7	45.4	43.7	53.1	+3.7
Query	71.5	59.5	53.3	61.5	59.0	56.7	55.9	46.1	39.8	52.0	+1.6
MI-Pruner [19] 2026.04	73.1	61.1	52.7	62.3	61.6	57.9	55.6	47.2	42.6	53.1	+3.7
VisionZip [58] CVPR'25	**74.2**	60.0	52.1	62.1	60.0	58.2	57.4	43.1	41.6	52.5	+2.5
FastVID [40] NeurIPS'25	73.0	59.9	51.7	61.5	61.2	58.4	58.0	44.4	41.8	52.8	+3.1
FlashVID [5] ICLR'26	**74.2**	60.8	52.2	62.4	61.4	**58.6**	**58.9**	45.0	41.9	53.4	+4.3
EchoPrune	73.0	**62.3**	**53.4**	**62.9**	**62.0**	**58.6**	57.8	**48.3**	**47.3**	**55.0**	**+7.4**
F160 (10×) / 10%											
Random	70.9	60.4	52.1	61.1	59.0	57.7	55.1	45.1	43.0	52.4	+2.3
Query	69.2	61.8	52.7	61.2	59.2	56.9	54.8	44.6	41.0	51.7	+1.0
MI-Pruner [19] 2026.04	71.2	62.4	53.4	62.4	61.6	58.2	56.4	46.0	39.3	52.5	+2.5
VisionZip [58] CVPR'25	70.7	60.1	**53.9**	61.6	**61.8**	**59.6**	56.8	45.1	40.2	52.7	+2.9
FastVID [40] NeurIPS'25	71.2	60.6	53.8	61.9	61.2	59.1	58.0	43.8	40.9	52.7	+2.9
FlashVID [5] ICLR'26	71.4	62.2	53.7	62.4	61.2	59.5	**58.9**	47.5	41.6	54.0	+5.5
EchoPrune	**72.7**	**63.2**	53.3	**63.1**	60.6	58.2	58.0	**50.6**	**47.9**	**55.6**	**+8.6**
F320 (20×) / 5% (ultra)											
Random	71.2	59.9	51.8	60.0	60.1	57.9	58.1	43.9	42.2	52.4	+2.3
MI-Pruner [19] 2026.04	69.9	61.8	51.9	61.2	61.8	57.6	53.9	46.6	39.9	51.8	+1.2
VisionZip [58] CVPR'25	68.7	61.1	51.5	60.4	61.9	58.1	54.2	47.0	40.6	52.1	+1.8
FastVID [40] NeurIPS'25	70.1	**62.0**	52.1	61.4	**62.2**	58.3	58.1	42.7	40.0	52.1	+1.8
FlashVID [5] ICLR'26	**72.7**	61.7	53.4	62.6	60.2	58.3	58.9	44.1	38.0	52.4	+2.3
EchoPrune	72.2	61.4	**54.8**	**62.8**	**62.2**	**58.7**	**59.0**	**51.4**	**46.3**	**55.6**	**+8.6**

4.2 Results

LLaVA-OV. As shown in Tab. 1, LLaVA-OV benefits from increased #frames (# denotes number), especially on long video benchmarks. For instance, LVBench consists of extremely long videos with an average duration of 4,101 seconds. As a result, higher sampling rates are necessary for its dense actions and long sequence processing. Under the setting of F96/33.3%, EchoPrune boosts the scores by 7.6%$^{\uparrow}$ on LVBench and 3.3%$^{\uparrow}$ on average. When pruning under 256 frames, EchoPrune achieves 105% performance compared with the baseline. On VideoMME-Long and MLVU, EchoPrune consistently improves the performance when more frames are available. Notably, two generic methods (VisionZip and MI-Pruner) maintain good performance on short videos in VideoMME-Short and Egoschema, while failing to identify pivot tokens in long videos (LongVideoBench, MLVU and LVBench). In the ultra compression case, we keep merely 10% tokens (56,448$^{\downarrow}$) from 320 frames (62,720 tokens). Under this setting, MI-Pruner, VisionZip and FastVID fail to improve the overall performance, and FastVID even decreases the average score by 7.5%$^{\downarrow}$. Compared with other pruning methods, EchoPrune achieves SOTA performance in most datasets while bypassing the inefficient segment-and-merge paradigm. Efficiency analysis is reported in Tab. 4.

Qwen2.5VL and Qwen3VL. Tab. 2 sets 16 frames as the baseline following FlashVID [5], then shows pruning effects under 4$\sim$20$\times$ frames. Despite the crossmodal importance, *Query* alone (implemented by Eqn. (4)) is a noisy importance estimator in token pruning. In short videos, it performs even worse than random pruning. This explains why most of the previous work advocates vision-based pruning, while none of them explore pure crossmodal scores. Yet, we claim that the synergy of crossmodal relevance and visual redundancy is necessary for optimal token allocation. Despite halved token budgets compared with Tab. 1, EchoPrune demonstrates comparable or even higher scores on 320 frames. Tab. 3 shows pruning results on Qwen3VL. We report two scales, 2B and 8B, to prove our general usage. Fig. 4 visualizes the retained slots in a video after our compression. Benefited from the semantic reconstruction, EchoPrune captures the second "`Guitar`" in different colors (top), and yields good OCR capacity (bottom). The logo "`smule`" and context cues "`FREE`, `subscribe`" are retained while the white wall background is masked. See more visualization in the appendix B.4.

Table 3: **Performance on Qwen3VL-series.** Setting F32 as the baseline, different pruning methods are compared under $5\times$ frame rates and the same token budgets.

Method	Setting	VideoMME				VideoMMMU*				LongVideo Bench	MLVU
		Short	Medium	Long	Overall	Per.	Comp.	Adap.	Overall		
Qwen3VL-8B	F32	75.0	59.8	56.6	63.8	73.0	64.0	35.7	57.6	58.1	46.2
MI-Pruner [19] 2026.04	*F160/20%*	**77.4**	66.8	**57.1**	**67.1**	67.7	55.0	28.3	50.3	60.7	53.5
VisionZip [58] CVPR'25		77.2	65.9	56.5	66.5	68.1	56.2	28.4	50.9	60.8	53.1
FlashVID [5] ICLR'26		77.0	65.7	55.1	65.9	71.7	58.7	36.3	55.6	**61.6**	**53.8**
EchoPrune		75.2	**67.8**	**57.1**	66.7	**74.3**	**61.3**	**38.3**	**58.0**	59.9	**53.8**
Qwen3VL-2B	F32	68.3	52.9	48.6	56.6	55.0	43.0	26.0	41.3	51.6	42.3
MI-Pruner [19] 2026.04	*F160/20%*	72.6	59.0	**47.1**	59.6	48.7	35.3	26.7	36.9	**54.8**	47.4
VisionZip [58] CVPR'25		72.8	59.9	47.0	59.9	48.6	36.6	27.1	37.4	54.3	47.1
FlashVID [5] ICLR'26		72.1	57.0	47.0	58.7	45.0	37.3	22.0	34.8	52.6	45.3
EchoPrune		**73.1**	**59.2**	46.9	**59.8**	**56.7**	**40.7**	**27.7**	**41.7**	**54.8**	**47.6**

* Per./Comp./Adap. of VideoMMMU represent Perception/Comprehension/Adaptation splits, respectively.

Table 4: **Efficiency on LLaVA-OV-7B.** We report the compression time, prefilling time and Time-To-First-Token (TTFT) in milliseconds. Our speedup on prefilling and TTFT is annotated in (N×).

Method	Settings	Compress.	Prefilling	TTFT	Scores	Settings	Compress.	Prefilling	TTFT	Scores
LLaVA-OV-7B	F64	-	917.2 (1×)	1569.4 (1×)	58.8	F160	-	2760.8 (1×)	4391.6 (1×)	58.6
HoliTom [39] NeurIPS'25	*F64/20%*	280.7	523.7	1175.9	58.4	*F160/20%*	830.4	1423.0	3053.8	60.1
UniComp [62] CVPR'26		210.1	353.5	1005.7	58.6		511.2	818.8	2449.6	60.2
FlashVID [5] ICLR'26		92.1	258.0	910.2	58.5		180.3	624.4	2255.2	60.0
EchoPrune		18.2	192.9 (4.8×)	845.1 (1.9×)	58.7		42.8	491.3 (5.6×)	2122.1 (2.1×)	60.4

Efficiency analysis. Tab. 4 assesses the efficiency of LLaVA-OV-7B. We report average values of VideoMME after warmups. The Time-To-First-Token (TTFT) consists of ViT encoding and prefilling (compression and LLM forward). Since all pruning methods share the same ViT, we report the compression costs, overall prefilling time and TTFT in Tab. 4. Due to the limited capacity, increasing frames to F160 doesn't boost the baseline performance (58.8 to 58.6). In comparison, our pruning reduces the #tokens to the same as the default F32 setting, improving the score to 60.4. On these two settings, EchoPrune accelerates around $5\times^{\uparrow}$ on prefilling time and $2\times^{\uparrow}$ on TTFT.

4.3 Ablation Study

Fig. 5 shows ablation study on VideoMME (F160) for LLaVA-OV-7B (top) and Qwen2.5VL-7B (bottom). See discussions on Qwen3VL in App. B.3. Fig. 5(left) contrasts various configurations of τ and ϵ. We select temperature τ=0.1 and window size ϵ=3 for LLaVA-OV-7B, and holistic matching with 0.5τ for QwenVL-series. Fig. 5(middle) illustrates the effectiveness of our global Top-K ranking and causal reconstruction. Specifically, the *uniform* setting replaces the global Top-K with per-frame Top-K under a uniform frame budget, which leads to diluted importance. The *reverse* defines δ_{echo} as the backward error (from frame k+1), and *bidirection* averages over forward (from k−1) and backward (from k+1) errors. Both yield inferior performance compared to our forward strategy, likely due to our preservation of temporal causality. Finally, our synergy design consisting

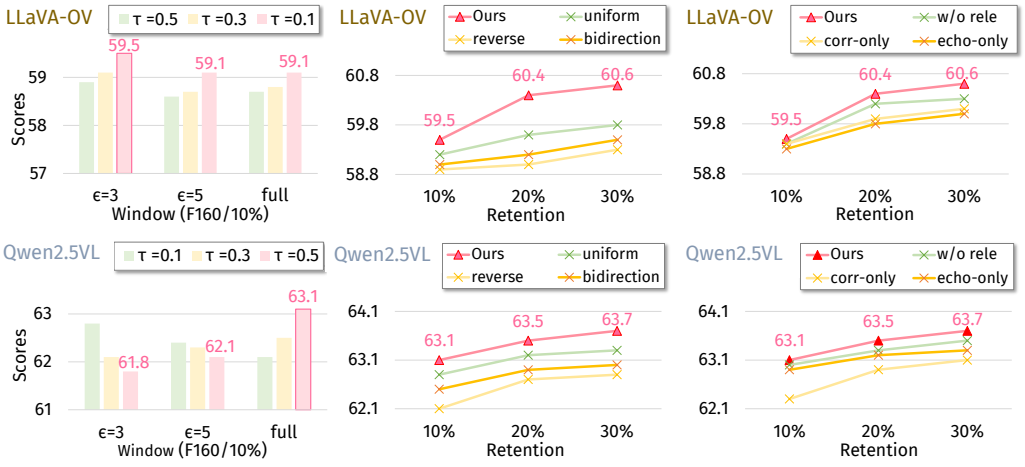

Figure 5: **Ablation study. Left:** LLaVA-OV performs best on 0.1τ and ϵ=3, while Qwen2.5VL suits 0.5τ and full-frame matching. **Middle:** Our global Top-K and causal δ is better than uniform, reverse and bidirectional designs. **Right:** The synergy yields SOTA results compared with single types.

of correspondence and echo matching is better than the one-type pattern, as reported in Fig. 5(right). The echo matching only (*echo-only*) works better than correspondence matching only (*corr-only*) in Qwen2.5VL for its stronger ViT encoder. And *w/o rele* removes the crossmodal relevance term (Eqn. (4)) for purely redundancy-guided ranking, which undermines the overall performance. See more discussions about the range of frame matching in App. B.3.

5 Related Work

VideoLLMs. The advanced multimodal language models have a strong video understanding capacity [2, 53]. Following the `Enc-MLP-Dec` architecture [27], LLaVA-OV [18] improves the MLLM performance in single-image, multi-image and video scenarios. The latest Qwen2.5VL [48] and Qwen3VL [2] further support adaptive resolution with stronger perception and comprehension performance. Despite the progress in VideoLLMs, their substantial GPU memory consumption and high inference latency pose critical bottlenecks for real-world deployment [26, 30]. Consequently, a growing body of research has diverted its focus toward efficient inference techniques [40, 46], encompassing both training-based and training-free paradigms.

Training-based token compression. Within the realm of training-based methods, the series of Mixture-of-Depths (MoD) [30, 65] employ a learnable router in LLMs to identify essential vision tokens while bypassing redundant ones. In parallel, low-resolution (LR) visual inputs naturally require fewer tokens and suffice for addressing coarse-grained queries. Recent studies [26, 59] advocate resolution-on-demand, which frames the policy of *whether to request high resolution* through the lens of reinforcement learning (RL). Formulating reconstruction error as token uniqueness, AutoGaze [41] incorporates VideoMAE [50] to perform autoregressive reconstruction before ViTs. Likewise, STTS [64] trains a scorer in ViTs with a reconstruction loss. Yet, the pruning on the visual encoder side needs extra tuning for multimodal alignment.

Training-free token compression. The training-free token reduction techniques are notable for their versatility and efficiency. The *generic methods* are applicable to both image and video pruning. For instance, VisionZip [58] and so on [67, 69] leverage attention matrices to identify salient tokens, and other methods analyze token similarities and coverages under a subset selection framework [1, 19, 68]. A conventional pipeline adopted by many *video-specific* methods [39, 40, 62] is characterized as segment-and-merging. Alternatively, FlashVID [5] and ForestPrune [16] organize tokens into tree-based structures for spatiotemporal modeling. Recent image pruning approaches [22, 33] leverage the *reconstruction error* as an importance measure, since unique tokens are not predictable [22, 51]. This principle is also extended to video compression [41, 64]. Yet, reconstruction-based pruning in a training-free manner remains relatively underexplored. To bridge the gap, we introduce temporal echo reconstruction and query-aware retention. Please refer to App. A.1 for a wider discussion.

6 Conclusion

In this paper, we propose a training-free video pruning method for efficient VideoLLMs. Motivated by maximum marginal relevance, ***EchoPrune*** employs query-guided *crossmodal relevance* and *temporal reconstruction error* for explicit spatiotemporal modeling. Working as a plug-in module, our method supports once-for-all compression before LLMs, bypassing segment and merging operations. Extensive experiments on LLaVA-OV, Qwen2.5VL and Qwen3VL across six benchmarks demonstrate our SOTA performance and efficiency. Under a fixed token budget, our method expands the processable frame rate to 20× with an average 7%$^{\uparrow}$ performance gain. Furthermore, ***EchoPrune*** accelerates the prefilling by 5.6× and TTFT by 2.1× on LLaVA-OV, which enables fine-grained long-term video understanding and low-latency responses in computation-limited scenarios.

Limitations and future work. Despite the benefits, there exist three drawbacks in our method. *(i)* For simplicity, we prune tokens once in the projection layer, without ViT acceleration strategies [60]. *(ii)* Our method retains the original indices for Rotary Position Embedding (RoPE). We believe that enhanced RoPE will boost the performance further [9]. *(iii)* We only consider the vision modality, regardless of auditory streams. The audio-guided video pruning [47] for Omnimodal Large Language Models becomes future work.

Acknowledgments and Disclosure of Funding

This research received funding from the Flemish Government (AI Research Program) and the Research Foundation Flanders (FWO) through project number G0G2921N. We acknowledge EuroHPC JU for awarding access to the EuroHPC supercomputer LEONARDO, hosted by CINECA (Italy) and the LEONARDO consortium. We thank Han Zhou for helpful discussions.

References

[1] Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In CVPR, 2025.

[2] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. arXiv preprint arXiv:2511.21631, 2025.

[3] Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998.

[4] Junhao Du, Jialong Xue, Anqi Li, Jincheng Dai, and Guo Lu. Unified spatiotemporal token compression for video-llms at ultra-low retention. In CVPR, 2026.

[5] Ziyang Fan, Keyu Chen, Ruilong Xing, Yulin Li, Li Jiang, and Zhuotao Tian. Flashvid: Efficient video large language models via training-free tree-based spatiotemporal token merging. In ICLR, 2026.

[6] Chaoyou Fu, Yuhan Dai, Yongdong 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. In CVPR, 2025.

[7] Tianyu Fu, Tengxuan Liu, Qinghao Han, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. Framefusion: Combining similarity and importance for video token reduction on large vision language models. In ICCV, 2025.

[8] Satoru Fujishige. Submodular functions and optimization, volume 58. Elsevier, 2005.

[9] Chao Gong, Depeng Wang, Zhipeng Wei, Ya Guo, Huijia Zhu, and Jingjing Chen. Echoingpixels: Aliasing-resistant joint token reduction for audio-visual llms. In ICML, 2026.

[10] Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Yuanhan Zhang, Xiang Yue, Bo Li, and Ziwei Liu. Video-mmmu: Evaluating knowledge acquisition from multi-discipline professional videos. arXiv preprint arXiv:2501.13826, 2025.

[11] Haifeng Huang and Yang Li. Kitoke: Kernel-based interval-aware token compression for video large language models. arXiv preprint arXiv:2604.03414, 2026.

[12] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In CVPR, pages 6700–6709, 2019.

[13] Yicheng Ji, Jun Zhang, Heming Xia, Jinpeng Chen, Lidan Shou, Gang Chen, and Huan Li. Specvlm: Enhancing speculative decoding of video llms via verifier-guided token pruning. In EMNLP, 2025.

[14] Yicheng Ji, Jun Zhang, Jinpeng Chen, Cong Wang, Lidan Shou, Gang Chen, and Huan Li. See the forest for the trees: Loosely speculative decoding via visual-semantic guidance for efficient inference of video llms. In ACL, 2026.

[15] Xin Jin, Jinming Liu, Yuntao Wei, Junyan Lin, Zhicheng Wang, Jianguo Huang, Xudong Yang, Yanxiao Liu, and Wenjun Zeng. Compression tells intelligence: Visual coding, visual token technology, and the unification. arXiv preprint arXiv:2601.20742, 2026.

[16] Shaobo Ju, Baiyang Song, Tao Chen, Jiapeng Zhang, Qiong Wu, Chao Chang, HuaiXi Wang, Yiyi Zhou, and Rongrong Ji. Forestprune: High-ratio visual token compression for video multimodal large language models via spatial-temporal forest modeling. In CVPR Findings, 2026.

[17] Quan Kong, Yuhao Shen, Yicheng Ji, Huan Li, and Cong Wang. Parallelvlm: Lossless video-llm acceleration with visual alignment aware parallel speculative decoding. In CVPR, 2026.

[18] 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.

[19] Jiameng Li, Aleksei Tiulpin, and Matthew B. Blaschko. MI-pruner: Crossmodal mutual information-guided token pruner for efficient mllms. arXiv preprint arXiv:2604.03072, 2026.

[20] Jiaqi Li, Shuntian Zheng, Yixian Shen, Jia-Hong Huang, Xiaoman Lu, Minzhe Ni, and Yu Guan. Keeping the evidence chain: Semantic evidence allocation for training-free token pruning in video temporal grounding. In ECCV, 2026.

[21] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In ICML, 2022.

[22] Xu Li, Yi Zheng, Yuxuan Liang, Zhe Liu, Xiaolei Chen, Haotian Chen, Rui Zhu, and Xiangyang Xue. Resprune: Text-conditioned subspace reconstruction for visual token pruning in large vision-language models. arXiv preprint arXiv:2603.21105, 2026.

[23] Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE: Speculative sampling requires rethinking feature uncertainty. In ICML, 2024.

[24] Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE-2: Faster inference of language models with dynamic draft trees. In EMNLP, 2024.

[25] Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE-3: Scaling up inference acceleration of large language models via training-time test. In NeurIPS, 2025.

[26] Huanxuan Liao, Zhongtao Jiang, Yupu Hao, Yuqiao Tan, Shizhu He, Jun Zhao, Kun Xu, and Kang Liu. Resadapt: Adaptive resolution for efficient multimodal reasoning. arXiv preprint arXiv:2603.28610, 2026.

[27] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In CVPR, 2024.

[28] Xuyang Liu, Yiyu Wang, Junpeng Ma, and Linfeng Zhang. Video compression commander: Plug-and-play inference acceleration for video large language models. In EMNLP, 2025.

[29] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In NeurIPS, 2022.

[30] Yaxin Luo, Gen Luo, Jiayi Ji, Yiyi Zhou, Xiaoshuai Sun, Zhiqiang Shen, and Rongrong Ji. $\gamma-$mod: Exploring mixture-of-depth adaptation for multimodal large language models. In ICLR, 2025.

[31] Junpeng Ma, Qizhe Zhang, Ming Lu, Zhibin Wang, Qiang Zhou, Jun Song, and Shanghang Zhang. Mmg-vid: Maximizing marginal gains at segment-level and token-level for efficient video llms. In AAAI, 2026.

[32] Junpeng Ma, Sashuai Zhou, Guanghao Li, Xin Gao, Yue Cao, Hengyu Zeng, Yuxiang Yan, Zhibin Wang, Jun Song, Bo Zheng, et al. Gift: Global irreplaceability frame targeting for efficient video understanding. In CVPR, 2026.

[33] Qiankun Ma, Ziyao Zhang, Haofei Wang, Jie Chen, Zhen Song, and Hairong Zheng. Apet: Approximation-error guided token compression for efficient vlms. In CVPR, 2026.

[34] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In NeurIPS, 2023.

[35] Harry Nyquist. Certain topics in telegraph transmission theory. Transactions of the American Institute of Electrical Engineers, 47(2):617–644, 1928.

[36] Tianyuan Qu, Longxiang Tang, Bohao Peng, Senqiao Yang, Bei Yu, and Jiaya Jia. Does your vision-language model get lost in the long video sampling dilemma? In ICCV, 2025.

[37] Alex Rodriguez and Alessandro Laio. Clustering by fast search and find of density peaks. science, 344(6191):1492–1496, 2014.

[38] Sayan Deb Sarkar, Rémi Pautrat, Ondrej Miksik, Marc Pollefeys, Iro Armeni, Mahdi Rad, and Mihai Dusmanu. Cope-videolm: Leveraging codec primitives for efficient video language modeling. arXiv preprint arXiv:2602.13191, 2026.

[39] Kele Shao, Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. Holitom: Holistic token merging for fast video large language models. In NeurIPS, 2025.

[40] Leqi Shen, Guoqiang Gong, Tao He, Yifeng Zhang, pengzhang liu, Sicheng Zhao, and Guiguang Ding. FastVID: Dynamic density pruning for fast video large language models. In NeurIPS, 2025.

[41] Baifeng Shi, Stephanie Fu, Long Lian, Hanrong Ye, David Eigen, Aaron Reite, Boyi Li, Jan Kautz, Song Han, David M Chan, et al. Attend before attention: Efficient and scalable video understanding via autoregressive gazing. In CVPR, 2026.

[42] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In CVPR, pages 8317–8326, 2019.

[43] Enxin Song, Wenhao Chai, Tian Ye, Jenq-Neng Hwang, Xi Li, and Gaoang Wang. Moviechat+: Question-aware sparse memory for long video question answering. TPAMI, 2025.

[44] Feilong Tang, Xiang An, Yunyao Yan, Yin Xie, Bin Qin, Kaicheng Yang, Yifei Shen, Yuanhan Zhang, Chunyuan Li, Shikun Feng, et al. Onevision-encoder: Codec-aligned sparsity as a foundational principle for multimodal intelligence. arXiv preprint arXiv:2602.08683, 2026.

[45] Xi Tang, Jihao Qiu, Lingxi Xie, Yunjie Tian, Jianbin Jiao, and Qixiang Ye. Adaptive keyframe sampling for long video understanding. In CVPR, 2025.

[46] Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. Dycoke: Dynamic compression of tokens for fast video large language models. In CVPR, 2025.

[47] Keda Tao, Kele Shao, Bohan Yu, Weiqiang Wang, Jian Liu, and Huan Wang. Omnizip: Audio-guided dynamic token compression for fast omnimodal large language models. In CVPR, 2026.

[48] Qwen Team. Qwen2.5-vl, January 2025. URL `https://qwenlm.github.io/blog/qwen2.5-vl/`.

[49] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. arXiv preprint physics/0004057, 2000.

[50] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In NeurIPS, 2022.

[51] Nan Wang, Zhiwei Jin, Chen Chen, and Haonan Lu. Pixelprune: Pixel-level adaptive visual token reduction via predictive coding. arXiv preprint arXiv:2604.00886, 2026.

[52] Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, et al. Lvbench: An extreme long video understanding benchmark. In ICCV, 2025.

[53] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265, 2025.

[54] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. In NeurIPS, 2024.

[55] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In ICLR, 2024.

[56] Zipeng Yan, Yinjie Chen, Chong Zhou, Bo Dai, and Andrew Luo. Vision transformers with self-distilled registers. In NeurIPS, 2025.

[57] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. arXiv preprint arXiv:2407.10671, 2024.

[58] Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. In CVPR, 2025.

[59] Senqiao Yang, Junyi Li, Xin Lai, Bei Yu, Hengshuang Zhao, and Jiaya Jia. Visionthink: Smart and efficient vision language model via reinforcement learning. In NeurIPS, 2025.

[60] Hanxun Yu, Wentong Li, Xuan Qu, Song Wang, Junbo Chen, and Jianke Zhu. Visiontrim: Unified vision token compression for training-free mllm acceleration. In ICLR, 2026.

[61] Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration. In ICML, 2024.

[62] Chao Yuan, Shimin Chen, Minliang Lin, Limeng Qiao, Guanglu Wan, and Lin Ma. Unicomp: Rethinking video compression through informational uniqueness. In CVPR, 2026.

[63] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In ICCV, 2023.

[64] Jianrui Zhang, Yue Yang, Rohun Tripathi, Winson Han, Ranjay Krishna, Christopher Clark, Yong Jae Lee, and Sangho Lee. Unified spatio-temporal token scoring for efficient video vlms. arXiv preprint arXiv:2603.18004, 2026.

[65] Jun Zhang, Desen Meng, Zhengming Zhang, Zhenpeng Huang, Tao Wu, and Limin Wang. p-mod: Building mixture-of-depths mllms via progressive ratio decay. In ICCV, 2025.

[66] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. In Findings of the Association for Computational Linguistics: NAACL 2025, 2025.

[67] Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond text-visual attention: Exploiting visual cues for effective token pruning in vlms. In ICCV, 2025.

[68] Qizhe Zhang, Mengzhen Liu, Lichen Li, Ming Lu, Yuan Zhang, Junwen Pan, Qi She, and Shanghang Zhang. Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. In NeurIPS, 2025.

[69] Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. In ICML, 2025.

[70] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, et al. Mlvu: Benchmarking multi-task long video understanding. In CVPR, 2025.

Appendix of EchoPrune: Interpreting Redundancy as Temporal Echoes for Efficient VideoLLMs

A Extended Related Work

A.1 Efficient VideoLLMs

We extend the technical background to demonstrate why training-free token pruning is a promising avenue for achieving efficient VideoLLMs. As mentioned in Sec. 1, the **Mixture-of-Depth** (MoD) mechanism operates on each LLM layer to select essential vision tokens for processing while skipping redundant ones. Typically, their MoD modules [30, 65] learn to assess the token importance with limited training data. A recent work, p-MoD [65] proposes a progressive ratio decay from shallow to deep layers. Back to the raw representation, **codec-aware** methods like OneVision-Encoder[44] and CoPE-VideoLM [38], leverage the codec information (e.g., motion vectors and residual signals) from video data to encode what moves and what changes between frames, thus preserving the sparsity of the original stream. CoPE-VideoLM [38] encodes the motion vectors and residuals (by Δ-encoder), then concatenates them with the keyframe tokens (from vision encoder) as the visual inputs, which supports dense temporal coverage at a fraction of the typical token budgets and runtime. Another technique is **speculative decoding**, which leverages a small model to generate a bunch of tokens and a large model to do one-time validation. One of its well-known applications in LLMs is the EAGLE-series [23–25]. In VideoLLMs, SpecVLM [13] proposes vision-aware drafting, including attention-guided Top-P retention and spatial uniform sampling. The following work LVSpec [14] introduces vision-aware verifying, and ParallelVLM [17] employs parallel drafting and verifying. Beyond token operations, a distinct line of research focuses on **keyframe selection**, which replaces the uniform downsampling initially in VideoLLMs with adaptive algorithms. For instance, AKS [45] downsamples the raw video into 1 frame per second and employs BLIP-ITM [21] to conduct adaptive keyframe sampling until the frame budgets. Similarly, GIFT [32] starts from uniformly sampled 128 frames and utilizes SigLip [63] to filter out keyframes by irreplaceability and budget-aware refinement. These frame selectors work as a pre-filtering module, which feed selected frames into VideoLLMs with the help of external Vision Language Models. Despite the efficiency benefits of the above methods, their implementation and application are less flexible than training-free token pruning in practice. We study token pruning for its superior versatility and minimal intrusiveness of the pretrained model.

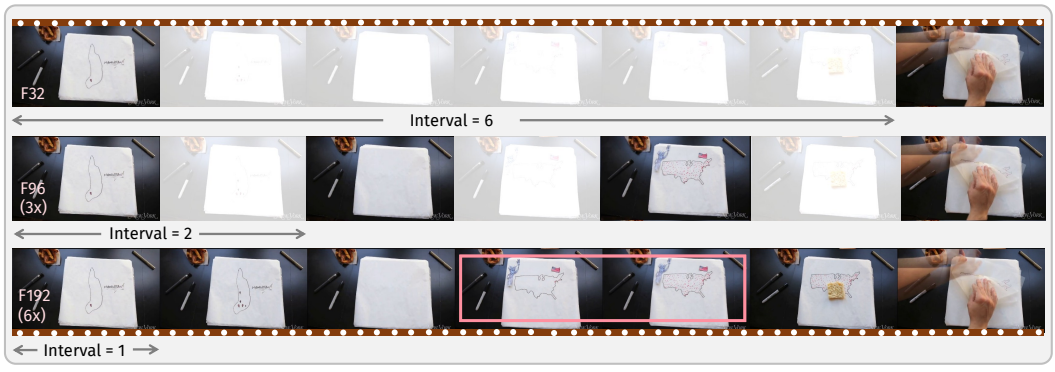

Figure 6: **Temporal resolution under different sampling rates.** The light-colored frames represent dropped clips under decreasing temporal resolution. Due to the large sampling interval, the pivot frames are fully skipped in the default F32 (the top row).

A.2 Rethinking Temporal Sampling and Aliasing Dilemma

The long-video sampling issue in VideoLLM benchmarks was first noticed by LSDBench [36]. It introduces the concept of *necessary sampling density*, defined as the minimum sampling density required to accurately answer a given question. Here, we discuss the undersampling bias mentioned in the main paper by reviewing the sampling interval (I) and temporal resolution (TR) from sampling theory. Specifically, the sampling interval refers to the stride between two consecutively sampled frames from a raw video sequence. Accordingly, temporal resolution characterizes the density of visual information (represented by the frame rate) captured per unit of time, which is the inverse of the sampling interval ($\mathrm{TR}{=}\frac{1}{I}$). When uniformly sampling M_F frames from M frames, the interval I and the temporal resolution TR are written as:

$$I = \frac{M}{M_F}; \tag{13}$$

$$\mathrm{TR} = \frac{1}{I} = \frac{M_F}{M}. \tag{14}$$

Fig. 6 treats 192 frames as the fully sampled baseline ($M{=}192$), *i.e.*, F192 has a sampling interval $I{=}1$ and temporal resolution $\mathrm{TR}{=}1$. As mentioned in Sec. 1, VideoLLMs uniformly sample several frames (denoted as M_F) from the raw video for the limited model capacity. To reduce M_F for faster inference, the interval increases to 2 with a halved temporal resolution ($\mathrm{TR}{=}\frac{1}{2}$), and the interval reaches 6 in a sparsely downsampled case F32 ($\mathrm{TR}{=}\frac{1}{6}$). We summarize three groups of I and TR in Tab. 5. The critical frames required to answer the query are highlighted in Fig. 6. Notably, F32 fail to capture these pivot moments due to its large sampling interval. As a result, the LLM is forced to generate responses based solely on pretrained priors rather than the actual visual evidence.

Table 5: Temporal sampling$_{M=192}$.

	F32	F96	F192
Interval (I)	6	2	1
Temporal resol. (TR)	$\frac{1}{6}$	$\frac{1}{2}$	1

According to Nyquist sampling theory [35], *a continuous signal can only be perfectly reconstructed if the sampling rate is at least twice the highest frequency component present in the signal*. Therefore, the aliasing occurs in F32 (Fig. 6, top), where the frame rate drops below the critical threshold, making lossless reconstruction mathematically impossible. Despite the reduced computation by processing fewer frames, the signal transmission in such a sparse-sampling scenario is fundamentally lossy. This prompts us to rethink the previous setting [39, 40] of pruning on F32 and comparing TTFT and FLOPs to show efficiency, since they overlook the unfairness caused by a "blinded LLM". To address it, our benchmark maintains the same token budget as the original F32 but scales the visible frames. From the view of signal processing, we increase the number of visible frames from M_F to $\gamma M_F (\gamma > 1)$ in order to satisfy the reconstruction threshold in Nyquist sampling theory. Then, we conduct pruning by temporal reconstruction to keep the high-frequency components.

As shown in Fig. 7, by preserving sufficient temporal resolution before pruning, we ensure that VideoLLMs "see" the essential content and then decide which tokens to proceed in decoding. Benefited from the temporal echo mechanism, EchoPrune selectively preserves the first pivot frame with high integrity, while the majority of the second pivot frame is pruned since it's reconstructable from the first frame. This behavior effectively demonstrates the interpretability of our method in identifying and removing temporal redundancy.

B Extended Experiments

B.1 Implementation Details

Efficiency analysis. The TTFT consists of ViT encoding, compression and LLM forward. In Tab. 4, we report the compression time, prefilling time (compression and LLM forward) and TTFT on a single A100 GPU by averaging over 100 samples in VideoMME. For each sample, we conduct 1 warmup and 3 runs to get the average time.

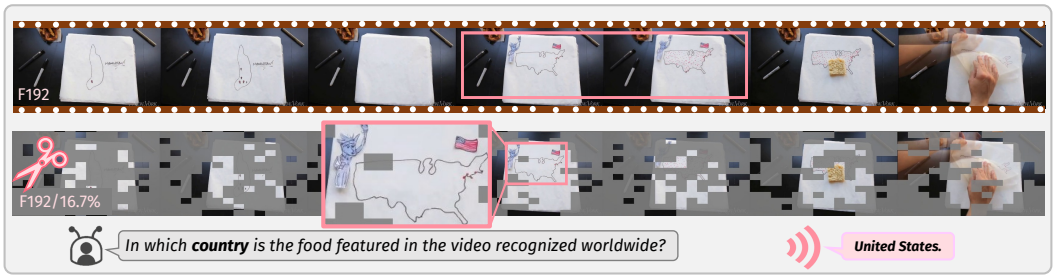

Figure 7: **Interpretability of EchoPrune (LLaVA-OV-7B).** To avoid undersampling, we first allow sufficient visible frames and then prune tokens (F192/16.7%) to reach the budgets of full-frame F32.

B.2 Dataset Description

VideoMME [6]. VideoMME comprises 900 videos spanning 6 diverse domains and 30 subcategories. The video durations range from 11 seconds to one hour, with an average duration of 1,018 seconds. Each video is accompanied by high-quality human annotations, including 2700 multiple-choice question-answer pairs.

EgoSchema [34]. EgoSchema includes 5031 multiple-choice questions derived from 250 hours of egocentric video. It emphasizes long-range temporal reasoning and poses substantial challenges for both fine-grained spatial perception and sustained temporal coherence. The subset is a reduced version of the full benchmark, with 500 questions.

LongVideoBench [54]. LongVideoBench consists of 3,763 videos from 8 seconds to 1 hour, and 6,678 human-annotated multiple-choice questions. The wide variation in video length and content makes it particularly challenging to evaluate temporal modeling, contextual understanding, and crossmodal alignment in video-language models. We tested on the validation set with 1337 questions.

MLVU [70]. MLVU comprises 3,102 multiple-choice questions spanning 9 long video understanding tasks. The videos range from 3 minutes to 2 hours, requiring reasoning over narrative structure, temporal order, and event retrieval. This benchmark jointly evaluates fine-grained spatial perception and long-range temporal reasoning capabilities. The test set includes 502 question-answer pairs.

VideoMMMU [10]. VideoMMMU evaluates the professional reasoning capacity in academic and scientific areas. The 300 expert-level videos have an average duration of 506 seconds, categorized into adaptation, comprehension and perception. It involves 900 multiple-choice questions.

LVBench [52]. LVBench is designed to challenge multimodal models' long-term memory and extended comprehension capabilities. Targeted at extremely long video understanding, LVBench contains 103 videos with an average duration of 4,101 seconds. To meet the demands of real-world applications, LVBench consists of 1549 manually annotated multiple-choice questions, spanning over six QA tasks.

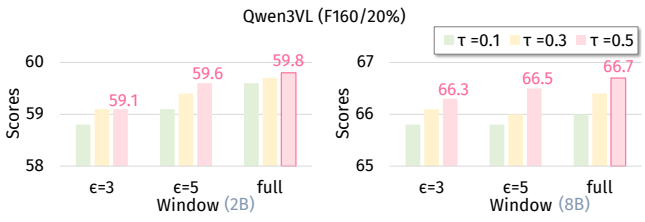

Figure 8: Ablation study on Qwen3VL.

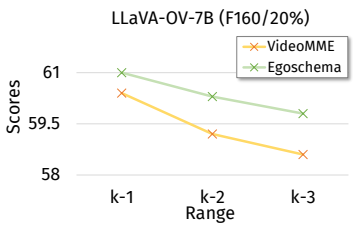

Figure 9: Various matching ranges.

B.3 Extended Evaluation

Ablation on Qwen3VL. Instead of per-dataset tuning, we adopt τ=0.5 and full-frame matching for QwenVL-series. We show the corresponding ablation study on Qwen2.5VL in Fig. 5, and extend the comparison on Qwen3VL-2B and 8B in Fig. 8 to justify our choice. In accordance with Qwen2.5VL, Qwen3VL exhibits SOTA performance on the same configuration (τ=0.5, full-frame). These empirical results demonstrate that our approach is user-friendly in terms of parameter configuration, requiring minimal effort for optimization.

Ablation on frame matching. We adopt the adjacent frame matching in Sec. 3.2, which enforces a localized temporal consistency constraint of the reconstruction error $\delta(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1})$. From the viewpoint of efficiency, we aim to avoid laborious iteration over all history frames from 1 to $k-1$. In Fig. 9, we provide empirical justification by extending the frame range to $k-2$ and $k-3$, including the nearest two frames and three frames, respectively. The range of $k-1$ involves tokens from the last frame (adopted by us), while the range of $k-2$ includes the last two frames with doubled candidate tokens, and the range of $k-3$ includes tripled. Empirically, the increased matching range undermines the performance due to the diluted temporal correspondence, which justifies our efficient choice.

Pruning under various FPS. In the main paper, we compare EchoPrune with SOTA methods under varying #frames. Here, we report Qwen2.5VL-7B performance under various frames per second (FPS). As shown in Fig. 10, we keep 20% tokens for both subset and total settings on Egoschema. EchoPrune consistently achieves SOTA performance on 1 FPS and 2 FPS, which demonstrates our robustness across various configurations.

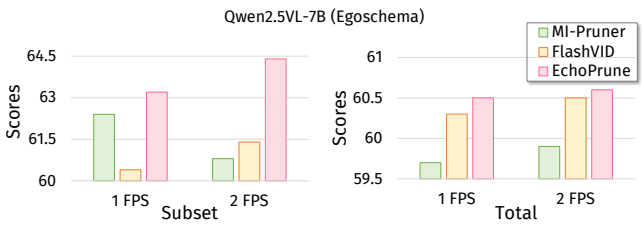

Figure 10: Performance under various FPS.

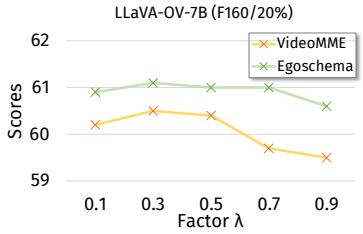

Figure 11: Various λ in MMR.

More model scales. Tab. 6 reports the pruning effects on 7B and 3B scales compared with three methods: random sampling, MI-Pruner [19] (generic pruning) and FlashVID [5] (video pruning). In the official repository, FlashVID sets the 20_{th} layer as the LLM pruning layer in Qwen2.5VL-7B. Since Qwen2.5VL-3B [48] has 36 LLM layers, we implement 25_{th} layer as the LLM pruning layer. The settings of MI-Pruner and EchoPrune are the same for 3B and 7B models. We conduct experiments on *F160/20%* settings. EchoPrune improves the average score by 5.7%$^{\uparrow}$ for the 7B model and 6.0%$^{\uparrow}$ for 3B, which shows our robustness.

Table 6: **Performance on Qwen2.5VL-7B and -3B.** Two baselines process 32 frames, while pruning methods are compared under 160 frames and 20% retained.

Method	VideoMME				EgoSchema		LongVideo Bench	MLVU	Video MMMU	Average	
	Short	Medium	Long	Overall	Subset	Total				Scores	%
32 frames (1×) / keep 100%											
Qwen2.5VL-7B	71.2	58.8	51.0	60.3	60.6	58.1	58.9	44.6	48.7	54.1	100
160 frames (5×) / keep 20%											
Random	74.4	63.9	52.3	63.6	61.0	59.0	60.5	49.7	42.3	55.0	+1.7
MI-Pruner [19] 2026.04	74.6	65.2	54.2	64.7	63.0	59.8	59.3	49.6	39.3	54.5	+0.7
FlashVID [5] ICLR'26	75.3	64.8	55.1	65.1	61.0	60.1	61.6	48.6	43.2	55.7	+3.0
EchoPrune	76.1	65.2	55.0	65.4	61.2	59.9	61.1	51.4	48.4	57.2	**+5.7**
32 frames (1×) / keep 100%											
Qwen2.5VL-3B	68.6	56.1	47.9	57.5	53.8	54.3	54.6	43.5	41.1	50.2	100
160 frames (5×) / keep 20%											
Random	70.3	60.1	48.3	59.6	55.2	55.0	55.1	48.0	40.1	51.6	+2.8
MI-Pruner [19] 2026.04	71.0	59.3	49.0	59.8	55.8	55.6	56.4	48.4	40.2	52.1	+3.8
FlashVID [5] ICLR'26	72.3	60.7	49.8	60.9	56.2	56.2	55.9	48.9	40.7	52.5	+4.6
EchoPrune	72.5	60.9	50.0	61.1	55.4	55.7	55.9	51.8	43.1	53.5	**+6.0**

MMR variants. In the main paper, we adopt a principled format of MMR to avoid introducing more hyperparameters, since both relevance and redundancy terms are derived by an inner product measure. Here, we recall Eqn. (11):

$$S(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}, \tilde{\mathcal{T}}) = r(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{T}}) - \delta(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1}). \tag{15}$$

and report more results when the relevance term is emphasized ($\lambda > 0.5$) or vice versa ($\lambda < 0.5$):

$$S(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}, \tilde{\mathcal{T}}) = \lambda \cdot r(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{T}}) - (1-\lambda) \cdot \delta(\tilde{\mathbf{v}}_i^k, \tilde{\mathcal{V}}^{k-1}). \tag{16}$$

We conduct experiments on VideoMME and Egoschema under the setting of F160/20%. As shown in Fig. 11, various choices of λ don't exhibit a clear benefit. On the contrary, relying more on the relevance (*e.g.*, $\lambda = 0.9$) term is harmful. Therefore, we keep the principled format in Eqn. (11), which can be understood as λ=0.5.

Image pruning. By design, the video and image pruning are not bidirectionally compatible, *i.e.* video-specific methods struggle to adapt to the single-frame static image. In comparison, the echo reconstruction mechanism in our temporal modeling is also applicable in image token selection. As shown in Tab. 7, EchoPrune achieves competitive performance on three open-ended benchmarks: GQA [12], SQA_{Img} [29] and TextVQA [42], based on LLaVA-1.5-7B. The comparison methods include coverage-based MI-Pruner [19], attention-based VisionZip [58] and reconstruction-based ApET [33]. Specifically, MI-Pruner employs crossmodal and intra-modal mutual information as the scoring function, aligning with the Maximum Marginal Relevance framework as our approach. Similarly, ApET utilizes linear approximation in LLMs to filter out redundant tokens based on L_2 reconstruction error, aligning with our motivation. While the literature on image pruning is extensive, we specifically focus on these representative methods due to their close methodological relevance. For a fair comparison with ApET, we initiate the reconstruction from M=10 tokens. The temperature τ=0.1 is the same as LLaVA-OV in the main paper. As shown in the table, ***EchoPrune*** achieves the highest average scores and robust performance on both video and image pruning tasks.

Table 7: **Performance on LLaVA1.5-7B (Image-QA).** The vanilla model contains 576 tokens, which are compressed to 128 (22.2% retained) and 64 (11.1% retained).

Method	GQA	SQA_{Img}	TextVQA	Average	GQA	SQA_{Img}	TextVQA	Average
	keep 128				*keep 64*			
MI-Pruner [19] 2026.04	58.5	69.5	56.1	61.4	56.9	**69.8**	54.9	60.5
VisionZip [58]CVPR'25	57.6	68.9	56.8	61.1	55.1	69.0	55.5	59.9
ApET [33] CVPR'26	58.9	68.7	53.9	60.5	56.9	68.9	53.0	59.6
EchoPrune	**59.1**	**69.7**	**57.1**	**62.0**	**57.6**	68.5	**55.6**	**60.6**

B.4 Extended Visualization

Our reconstruction error is derived from the synergy of correspondence and echo matching. As described in Sec. 4.1, echo matching can be implemented in a neighbor region Ω_i^ϵ (for LLaVA-OV), and in full-frame (for Qwen2.5VL- and Qwen3VL-series). We illustrate different ranges of frame matching in Fig. 12. The choice of receptive field for echo matching is primarily dictated by the underlying vision encoder. LLaVA-OV adopts SigLip [63] and Qwen2 [57] as the vision encoder and



Figure 12: **Frame matching.** *Correspondence* (left) refers to the history patch in the same location. *Neighbor* (middle) and *Holistic* (right) of echo matching leverage the patches in a neighbor region or from the full frame, respectively.

LLM decoder. It needs more spatial constraints for its fixed resolution, *i.e.* 196 tokens per frame. As a result, we conduct echo matching in a neighborhood for spatial consistency. In comparison, Qwen2.5VL [48] introduces dynamic resolution processing and absolute time encoding, enabling it to process images of varying sizes and videos of extended durations. To provide more contextual information, we employ holistic echo matching for QwenVL-series.

We provide further visualizations of our adaptive pruning effects on LLaVA-OV in Fig. 13, featuring two distinct questions based on the same video clip. When querying specific landmarks, EchoPrune effectively tracks the visual targets across multiple frames. The second question requires fine-grained retrieval regarding a staff member's clothing pattern, which appears in four frames. By discarding predictable tokens with minimal reconstruction error, EchoPrune successfully prunes the redundant frames. Specifically, among the four frames containing the staff (second row), only the first and third are retained, which correspond to distinct moments when the staff's head turns left and right, respectively. This exhibits that our method captures the new event with unpredictable information.

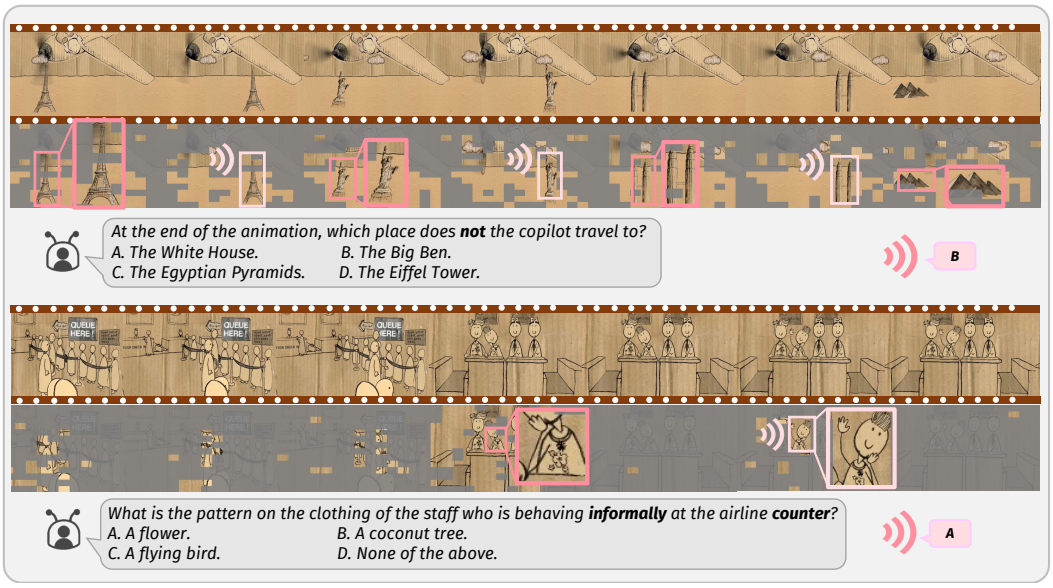

Figure 13: **Pruning effects on LLaVA-OV-7B** (VideoMME, *F160/20%*). Our pruning effectively eliminates query-irrelevant patches while preserving pivot slots. Specifically, we track landmarks consistently **(top)** and identify new events of varying head directions **(bottom)**.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

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

   Answer: [Yes].

   Justification: We aims at efficient MLLMs, and introduce crossmodal relevance and temporal reconstruction error to capture it.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes].

Justification: As shown in the final section, we state that extending to Omnimodal LLMs and so on are our limitations, which are also our future work.

Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [N/A].

   Justification: We don't propose any theory.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We describe the implementation and general setups in Sec. 4.1. The code is released on GitHub.

   Guidelines:

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

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

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

   Answer: [Yes]

   Justification: We conduct experiments on public datasets, and reproduce previous methods by their official repositories. The code of our method will also be released upon acceptance.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: As shown in Sec. 4.1, we describe the setup in detail. As a training-free method, we focus on the test settings.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: We don't report error bars, but show the improved percentage on average.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We conduct experiments on a single A100 GPU, as described in Sec. 4.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We conduct experiments on public dataset.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Please refer to our ethical statement.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

Justification: The paper will not release data or models with a risk of misuse.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: We appropriately cite the creators of all datasets, models, and code packages used in our paper.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

Answer: [N/A]

Justification: We do not release new assets.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: LLM is not an important, original or standard component of the core methods in our paper.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.