Sink-Token-Aware Pruning for Fine-Grained Video Understanding in Efficient Video LLMs

Kibum Kim[1,✉], Jiwan Kim[1,✉], Kyle Min[2,✉], Yueqi Wang[3,✉], Jinyoung Moon[4,✉], Julian McAuley[3,✉], and Chanyoung Park[1,✉]

[1]Korea Advanced Institute of Science and Technology (KAIST)
[2]Oracle
[3]University of California, San Diego
[4]Electronics and Telecommunications Research Institute (ETRI)

Abstract. Video Large Language Models (Video LLMs) incur high inference latency due to a large number of visual tokens provided to LLMs. To address this, training-free visual token pruning has emerged as a solution to reduce computational costs; however, existing methods are primarily validated on Multiple-Choice Question Answering (MCQA) benchmarks, where coarse-grained cues often suffice. In this work, we reveal that these methods suffer a sharp performance collapse on fine-grained understanding tasks requiring precise visual grounding, such as hallucination evaluation. To explore this gap, we conduct a systematic analysis and identify *sink tokens*–semantically uninformative tokens that attract excessive attention–as a key obstacle to fine-grained video understanding. When these sink tokens survive pruning, they distort the model's visual evidence and hinder fine-grained understanding. Motivated by these insights, we propose **S**ink-**To**ken-aware **P**runing (SToP), a simple yet effective plug-and-play method that introduces a sink score to quantify each token's tendency to behave as a sink and applies this score to existing spatial and temporal pruning methods to suppress them, thereby enhancing video understanding. To validate the effectiveness of SToP, we apply it to state-of-the-art pruning methods (VisionZip, FastVid, and Holitom) and evaluate it across diverse benchmarks covering hallucination, open-ended generation, compositional reasoning, and MCQA. Our results demonstrate that SToP significantly boosts performance, even when pruning up to 90% of visual tokens. Our code is available at https://github.com/rlqja1107/SToP

Keywords: Video LLMs · Visual Token Pruning · Attention Sink

1 Introduction

Video Large Language Models (Video LLMs) [2,20,22,25,50] have demonstrated their potential in video understanding by achieving remarkable performance in video-centric tasks, such as video question answering. However, they typically require processing thousands of visual tokens across multiple frames, leading to substantial inference latency and computational cost that hinders practical deployment. While previous studies [24, 30, 37, 47] have explored visual token

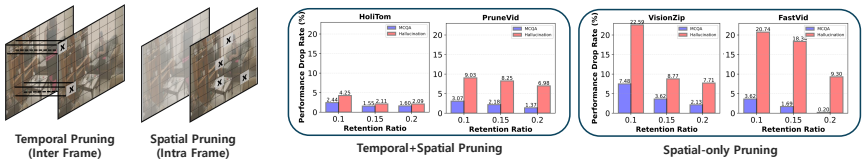

Fig. 1: (a) Overview of temporal and spatial pruning. (b) Performance drop rate relative to the vanilla model for temporal+spatial pruning and spatial-only pruning methods on the MCQA (MVBench [23]) and hallucination benchmark (EventHallusion [51]).

compression during training, these approaches often demand computationally expensive retraining.

To circumvent this, many recent studies [14, 36, 48, 53] adopt training-free visual token pruning, which compresses the visual tokens and reduces the number of tokens fed into the LLM without fine-tuning. In this regard, two main approaches, shown in Fig. 1(a), are commonly employed for visual token pruning: **Temporal pruning** (inter-frame) reduces redundancy along the temporal dimension by merging similar tokens at the same patch locations across frames, while **Spatial pruning** (intra-frame) identifies and retains salient tokens within each frame based on importance metrics (e.g., attention weights), merging the pruned tokens to preserve contextual information. Prior work on visual token pruning either combines temporal and spatial pruning (temporal+spatial pruning) [10, 14, 15, 35] or mainly focuses on spatial pruning (spatial-only pruning) [36, 48, 52, 53] to improve efficiency. For example, HoliTom [35] performs segment-wise temporal pruning followed by attention-based spatial pruning. On the other hand, FastVid [36] performs spatial pruning within each frame based on density and attention scores, without temporal pruning. Overall, these approaches facilitate significant inference acceleration by substantially reducing the visual token budget provided to the LLM.

However, despite their efficiency gains, most token pruning methods [35, 36, 48,52,53] are mainly evaluated on Multiple-Choice Question Answering (MCQA) benchmarks [12,23,31], where the models can often shortcut to the correct answer using language priors and coarse visual cues [4], potentially masking degradations in the underlying model's video understanding. In real-world applications, by contrast, users typically pose open-ended queries without predefined options in conversational scenarios [30], or require intricate spatio-temporal reasoning over open-world videos [18, 32], where correct answers must be grounded in fine-grained visual cues (e.g., subtle changes in actions and objects). In this regard, it is crucial to examine whether existing visual token pruning methods indeed preserve such fine-grained visual cues while reducing computation.

To explore this gap, we evaluate pruning methods through the lens of hallucination[1], which is highly sensitive to accurate visual evidence and thus serves as a proxy for how well fine-grained visual cues are preserved. Specifically, we measure the performance drop relative to an unpruned baseline (LLaVA-OneVision-7B [22]), which uses all visual tokens over 32 frames, under varying retention

[1] For example, prompts such as `Please describe the video in detail` demand strong visual grounding.

ratios on a hallucination dataset (EventHallusion [51]) and an MCQA dataset (MVBench [23]). For deeper analysis, we compare temporal+spatial pruning approach with spatial-only approach. As shown in Fig. 1(b), we observe that while both approaches remain relatively robust on the MCQA dataset, their performance collapses much more sharply on the hallucination dataset, indicating that existing pruning methods tend to be more vulnerable on tasks requiring fine-grained visual grounding than on MCQA. Moreover, temporal+spatial pruning is consistently more robust than spatial-only pruning, especially on the hallucination dataset, suggesting that the mechanisms underlying spatial pruning are particularly prone to hindering fine-grained video understanding.

Building on these findings, in Sec. 3, we analyze why spatial-only pruning and temporal+spatial pruning yields different levels of video understanding, and identify a key obstacle that hinders fine-grained video understanding. Specifically, we find that a small subset of visual tokens exhibit a *sink* phenomenon [16, 45], where attention is overly concentrated on regions that contain little semantic information. If such sink tokens survive pruning, the model is more likely to hallucinate because its generation is grounded in distorted visual evidence. Interestingly, although temporal pruning is primarily designed to remove inter-frame redundancy, it often suppresses sink tokens as a byproduct, which helps explain the greater robustness of temporal+spatial pruning.

Motivated by our analysis, we propose **S**ink-**To**ken-aware **P**runing (SToP), a training-free visual token pruning method that explicitly targets sink tokens during pruning. Specifically, we first define a *sink score* that quantifies each token's tendency to act as a sink. Based on this score, we design a Sink-Token-aware Spatial Pruning (STSP) module that adjusts the attention distribution of sink tokens, lowering their priority for retention—even when conventional attention-based importance scores would otherwise rank them highly. This design explicitly encourages the pruning of tokens that may appear salient under standard attention-based criteria but in fact correspond to sink behavior. In addition, we design a Sink-Token-aware Temporal Pruning (STTP) module, which further promotes the elimination of sink tokens along the time dimension.

To validate the effectiveness of SToP, we apply it to the state-of-the-art pruning methods (VisionZip [48], FastVid [36], and HoliTom [35]) and evaluate across multiple aspects of video understanding: hallucination evaluation (EventHallusion [51]), compositional reasoning (VideoComp [18]), and open-ended generation (VCG-Bench [30]). Our results demonstrate that SToP substantially enhances video understanding even under an extreme token budget (e.g., pruning 90% of tokens). These benefits also carry over to MCQA benchmarks, where we observe consistent performance gains.

We summarize our main contributions as follows: 1) Through a systematic analysis of visual token pruning, we identify that *sink tokens* are a key obstacle that hinders fine-grained video understanding. 2) We propose SToP, a training-free token pruning method that introduces a sink score and incorporates it into both spatial and temporal pruning to explicitly guide the pruning of sink tokens. 3) Extensive experiments across diverse video understanding tasks (e.g., hallucination, compositional reasoning, open-ended generation, and MCQA) demonstrate the effectiveness of SToP, consistently outperforming baselines.

2 Preliminary

2.1 Inference Complexity of Video LLMs

Architecture of Video LLMs. Given a user's question q and an input video of T frames, a visual encoder (e.g., SigLIP [49]) processes T frames independently to extract n_v patch embeddings per frame, followed by projecting them into the language embedding space with a neural network (e.g., MLP [22,26]). This yields visual tokens $H_v \in \mathbb{R}^{T \times n_v \times d}$, where n_v is the number of visual tokens per frame and d is the dimensionality of the LLM word embeddings. The user's question prompt is converted into text tokens $H_q \in \mathbb{R}^{n_q \times d}$, where n_q is the number of text tokens for q. The model then concatenates H_v and H_q and feeds the combined sequence into the LLM, which generates the response autoregressively by predicting the next text token at each decoding step.

Computational Complexity. We analyze the inference cost of Video LLMs in terms of Floating-Point Operations (FLOPs) and how the number of frames and visual tokens affects the overall computation. Given that the dominant cost arises from self-attention and feed-forward network (FFN) within the LLM, the total FLOPs across the LLM's L layers can be expressed as:

$$\text{FLOPs} = L \times (4nd^2 + 2n^2d + 2ndm), \tag{1}$$

where $n = T \cdot n_v + n_q$ is the input sequence length and m is the intermediate dimension of the FFN. It is important to note that the term $T \cdot n_v$ typically dominates n (e.g., $n_v = 6,270$ for 32 frames on LLaVA-OneVision [22], while $n_q \approx 20$). Furthermore, adding frames increases the sequence length by n_v per frame, incurring the attention cost quadratically through the $2n^2d$ term and the projection and FFN costs linearly through the $4nd^2 + 2ndm$ terms, with all costs further scaled by the depth L. This analysis highlights the importance of reducing the number of visual tokens $T \cdot n_v$ for efficient inference. In Sec. 5.2, we show that STOP remains robust on fine-grained video tasks (e.g., hallucination) even when pruning 90% of visual tokens (i.e., retaining only $0.1 \cdot T \cdot n_v$).

2.2 Token Pruning Approach

To reduce the number of visual tokens provided to LLMs, spatial [36,43,48] and temporal [10,35,41] pruning approaches are commonly used.

Spatial Pruning. Within each frame, it is important to retain the salient visual tokens while removing those that carry little semantic information. To this end, many token pruning methods [10,34,35,43,48,52] utilize the attention weights of a vision encoder, retaining tokens with high attention scores. Specifically, following prior work [35,36,43,48,52], we perform token selection using the CLS token. For visual encoders without an explicit CLS token (e.g., SigLIP [49]), we instead construct a CLS-equivalent attention [10,35,48]. More precisely, the attention matrix is computed as:

$$\hat{A} = \text{Softmax}(QK^T/\sqrt{d}) \in \mathbb{R}^{T \times n_v \times n_v}, \tag{2}$$

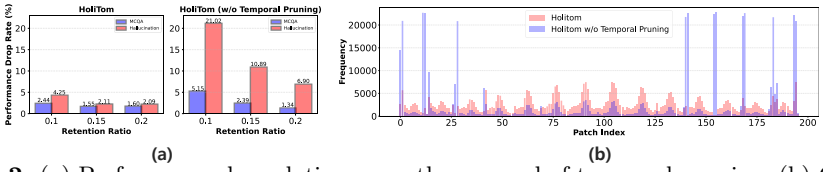

Fig. 2: (a) Performance degradation upon the removal of temporal pruning. (b) Comparison of selected visual token distributions between temporal+spatial pruning and the variant without temporal pruning as a retention ratio of 0.1. We use the EventHallusion [51] dataset for this analysis.

where Q and K are the query and key of the visual tokens, respectively. Following [10, 35, 48], a token's attention score is obtained by averaging the attention matrix column-wise, indicating that a token receiving more attention from other tokens is deemed more important. This yields an attention score matrix $A \in \mathbb{R}^{T \times n_v}$, which is used to select salient visual tokens with high attention score for each frame.

Temporal Pruning. In a video, some regions may stay unchanged over consecutive frames (e.g., a stationary object), resulting in temporal redundancy. To address it, prior work [14, 15, 35, 41] commonly detects redundancy by computing the similarity between tokens at the same patch location in adjacent frames, and then prunes tokens whose similarity exceeds a threshold. Formally, the patch indices pruned at frame t are defined as:

$$P_t = \{i | \text{sim}(H_v^{i,t}, H_v^{i,t+1}) > \tau, i \in [1, n_v]\}, \tag{3}$$

where τ is pruning threshold, $H_v^{i,t}$ denotes the token at the i-th patch location in the t-th frame, and $\text{sim}(\cdot, \cdot)$ is the cosine similarity between the two tokens. In practice, this pairwise similarity is often computed for every adjacent-frame pair within a temporally segmented clip and then aggregated (e.g., product); if the aggregated similarity exceeds a threshold, the tokens at the corresponding location within the clip are removed [14, 35, 41].

3 Identifying a Key Obstacle to Fine-Grained Video Understanding

In this section, we examine the gap in video understanding between temporal+spatial pruning and spatial-only pruning approaches, as shown in Fig. 1(b), and aim to identify a key obstacle to fine-grained video understanding. In line with experimental settings in Sec. 1, we conduct this study on the EventHallusion [51] dataset using LLaVA-OneVision [22].

3.1 Impact of Temporal Pruning for Fine-Grained Understanding

We hypothesize that the temporal pruning component helps preserve fine-grained visual cues beyond merely reducing temporal redundancy, as it is the core difference between the two approaches. To test this, we utilize Holitom [35], a temporal+spatial pruning method, as our baseline and evaluate the performance

drop when we intentionally remove temporal pruning.[2]. As shown in Fig. 2(a), we observe that removing temporal pruning substantially increases hallucinations, indicating that temporal pruning supports fine-grained video understanding, which we further discuss in Sec. 3.3. To uncover the cause of this drop, we analyze the distribution of selected visual tokens over the entire video. As shown in Fig. 2(b), we observe that without temporal pruning, a small subset of visual tokens is selected with disproportionately high frequency. This suggests that repeatedly selecting these tokens may harm fine-grained video understanding. Before validating it, we trace the source of these abnormal selection frequencies.

3.2 Source of High Frequency: Spatially Persistent High Attention

To characterize these high-frequency tokens identified in Fig. 2(b), we visualize the patch-wise attention scores over consecutive frames. As shown in Fig. 3, we observe that high attention scores persist at fixed spatial coordinates throughout the temporal sequence. Moreover, given that salient objects tend to appear near the center of the frame [42], these persistent high-attention regions are likely to correspond to semantically sparse background areas. Drawing a parallel in LLMs, where attention highly concentrates on semantically uninformative tokens at fixed positions (e.g., the BOS token) [40, 45], we refer to these spatially persistent high-attention tokens as ***sink tokens***[3]. Due to their high attention weights, sink tokens are preferentially retained during spatial pruning. In other words, their repeated retention at fixed coordinates leads to abnormally high selection frequencies (e.g., the lower-left patches in Fig. 3 with indices 154 and 155), observed in Fig. 2(b).

Fig. 3: Visualization of attention scores across consecutive frames.

3.3 *Sink Tokens* as an Obstacle to Fine-Grained Understanding

To verify whether the frequent selection of sink tokens directly hinders fine-grained video understanding and exacerbates hallucination, we perform a diagnostic experiment using VisionZip [48], a spatial-only pruning method that is relatively prone to hallucination (see Fig. 1). Specifically, to deliberately reduce the retention of sink tokens in a simple manner, we randomly remove sink tokens at varying ratios from the selected token set[4]. As shown in Fig. 4, we observe that as the removal ratio of

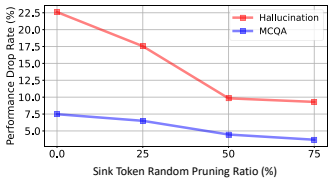

Fig. 4: Performance degradation rate across varying sink token removal ratios.

[2] To maintain a constant token budget when temporal pruning is removed, we proportionally increase the number of tokens selected during spatial pruning.

[3] Regarding a discussion of differences in perspectives on sink tokens in LLMs, please refer to Appendix A.

[4] To maintain a constant token budget, removed sink tokens are replaced with an equal number of tokens with the next highest attention scores.

sink tokens increases, performance degradation in hallucination diminishes significantly while improving MCQA performance. This indicates that sink tokens are directly detrimental to fine-grained video understanding. We hypothesize this is because sink tokens generally carry limited semantic information despite high attention scores. As a result, repeatedly retaining them consumes the limited token budget, leaving insufficient capacity for truly salient visual tokens and causing text generation based on distorted or incomplete evidence. Regarding a detailed ablation on the inherent impact of sink tokens, please refer to Sec. D.

Rethinking the Role of Temporal Pruning. While temporal pruning is originally designed to reduce temporal redundancy [10, 14, 35], we posit that it implicitly acts as a sink token suppressor. This is because sink tokens tend to reside in background regions, as discussed in Sec. 3.1, making their representations highly similar across adjacent frames and thus more likely to be removed by temporal pruning. In fact, when comparing the number of sink tokens[5] selected by Holitom (temporal+spatial pruning) against an variant without temporal pruning (spatial pruning), which is introduced in Sec. 3.1, we observe that temporal pruning reduces sink tokens by 83% (384 → 66). Coupled with the performance drop observed upon the removal of temporal pruning (Fig. 2(a)), this correlation further supports our hypothesis that the frequent selection of sink tokens is a key obstacle to fine-grained video understanding.

4 Proposed Method: Sink-Token-aware Pruning (STоP)

Building on our analysis in Sec. 3, we propose STоP, a plug-and-play token pruning method designed to explicitly guide the pruning of sink tokens. We begin by defining a sink score to quantify how likely each token is to behave as a sink token (Sec. 4.1). This score is then seamlessly incorporated into existing spatial and temporal frameworks [14, 35, 36, 48] via two proposed modules: the STSP module for spatial pruning and the STTP module for temporal pruning (Sec. 4.2).

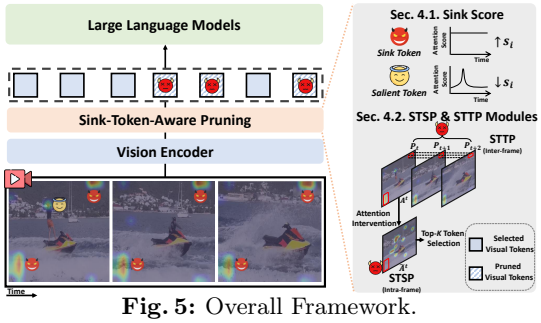

Fig. 5: Overall Framework.

4.1 Quantifying Token Persistence via Sink Score

While sink tokens can be broadly identified by the distribution gap between the temporal+spatial and spatial-only pruning (Fig. 2(b)), it remains challenging to finely distinguish sink tokens from truly salient tokens, as both exhibit high attention values. However, as discussed in Sec. 3.2, sink tokens are typically characterized by spatially persistent high attention across the temporal dimension.

[5] Herein, the set of sink tokens is defined as the top 10% highest-frequency tokens in Fig. 2(b).

To formalize this behavior, we define the sink score s_i for each visual token i as:

$$s_i = \text{MinMax-Norm}(\hat{s}_i^w), \hat{s}_i = \sum_{t=1}^{T} A_i^t, \tag{4}$$

where A_i^t denotes the attention score of the i-th token in the t-th frame ($i \in [1, n_v]$). Intuitively, $\hat{s}_i$ yields a high value when a token's attention remains consistently elevated across frames, identifying it as a likely sink token. On the other hand, non-sink tokens that are only transiently salient—scoring highly in only a few frames—result in a lower $\hat{s}_i$. Lastly, we apply min-max normalization (MinMax-Norm) to the power term $\hat{s}_i^w$ to map the values to the range $[0, 1]$. The hyperparameter w serves to sharpen the distribution of s_i, suppressing the influence to the non-sink tokens while effectively highlighting the sink tokens. Regarding a detailed analysis of w, please refer to Appendix E.

4.2 Sink-Token-aware Pruning

STSP: Sink-Token-aware Spatial Pruning. Leveraging the sink score s_i, we propose a plug-and-play spatial pruning method, termed **STSP** module, which explicitly attenuates the selection of sink tokens during pruning. Conventional spatial pruning methods [35, 36, 48] generally retain tokens with high attention scores, operating under the assumption that these scores directly correlate with visual importance. However, their methods often inadvertently select sink tokens—which exhibit high attention values despite a lack of semantics—thereby hindering fine-grained video understanding. To address this, we integrate the sink score into the attention mechanism as:

$$\tilde{A}_i^t = A_i^t - \mu_s \cdot s_i, \tag{5}$$

where μ_s is a hyperparameter controlling the influence of the sink score. By penalizing sink-prone tokens with high sink scores, we effectively reduce sink tokens' attention score, preventing the selection of semantically sparse tokens that would otherwise dominate the selection process. This intervention ensures that a higher proportion of genuinely salient tokens are selected, ultimately enhancing video understanding even at a low retention ratio.

STTP: Sink-Token-aware Temporal Pruning. Although temporal pruning implicitly discourages the retention of sink tokens, these tokens still survive after temporal pruning, as discussed in Sec. 3.3. To mitigate this, we propose **STTP** module, which explicitly leverages the sink score to further facilitate the pruning of sink tokens along the temporal axis. Formally, we integrate this score into Eq. (3) as follows:

$$P_t = \{i | \text{sim}(H_v^{i,t}, H_v^{i,t+1}) + \mu_t \cdot s_i > \tau, i \in [1, n_v]\}, \tag{6}$$

where μ_t is the hyperparameter to control the weight of s_i. The term $\mu_t \cdot s_i$ acts as a factor that pushes sink tokens above the threshold τ, thereby explicitly assigning them to the pruning set P_t.

5 Experiment

5.1 Experimental Setting

Dataset. To assess fine-grained visual grounding, we evaluate SToP across multiple dimensions of video understanding. Specifically, we utilize EventHallusion [51] to measure *hallucination* via binary questions (Binary), which probe susceptibility to language bias and therefore require fine-grained video understanding—and open-ended descriptive tasks (Desc.). We utilize VideoComp [18] to evaluate *compositional reasoning*, specifically focusing on complex temporal and structural relationships and subtle action transitions, and report results on two domains: ActivityNet (Act.) and YouCook2 (YC). Finally, to evaluate *open-ended generation*, we employ VCG-Bench [30]. For all free-form text generation evaluations (EvetHallusion-Desc., and VCG-Bench), we use GPT-5 [38] for scoring. Furthermore, we evaluate SToP on five widely-used MCQA benchmarks: MVBench [23], VideoMME [12], NextQA [46], LongVideoBench [44], and MLVU [55]. Additional details on these datasets and their metrics are described in Appendix F.1.

Baselines. SToP is agnostic to attention-based spatial pruning, allowing it to integrate seamlessly with various pruning methods. Specifically, we apply it to spatial pruning (VisionZip [48] FastVid [36]) equipped with the STSP module, and temporal+spatial pruning (Holitom [35]) equipped with the STSP and STTP modules. We compare these integrated versions against their respective baselines, and the state-of-the-art method, PruneVid [14]. To further demonstrate the adaptability of SToP, we apply it to FlashVid [10], the most recent visual token pruning method, described in Appendix F.2.

Implementation Details. Following the experimental setup [35], we implement our method on LLaVA-OneVision-7B [22] and LLaVA-Video-7B [54]. For all evaluations, we utilize 32 frames, resulting in a total of 32×196 visual tokens. In line with [10, 35], we evaluate our method using retention ratios of 10%, 15%, and 20%. Regarding hyperparameters, we set w to 1.1 across all experiments. The threshold parameter μ_s is set to 0.03 for VisionZip and 0.02 for FastVid. All models are evaluated on an NVIDIA GeForce A6000 48GB GPU. Unless otherwise stated, we conduct experiments using LLaVA-OneVision-7B as the primary backbone. Additional implementation details are provided in the Appendix F.3. Regarding experiments of hyperparameter sensitivity, please refer to Appendix F.4.

5.2 Main Results

We evaluate SToP by integrating it into three state-of-the-art methods: VisionZip, FastVid (spatial pruning), and Holitom (temporal+spatial pruning). Our evaluation spans fine-grained video tasks and MCQA benchmarks.

Results on Fine-Grained Video tasks. Tab. 1 shows the performance of SToP compared to baselines on fine-grained video tasks across the EventHallusion, VideoComp, and VCG-Bench datasets. We have the following observations:

Table 1: Performance comparison on benchmarks requiring fine-grained video understanding. Performance Drop Rate denotes the macro-average of performance degradation relative to the Vanilla model across all categories. Desc., Consist., Temp., Act., YC are Description, Consistency, Temporal, ActivityNet, and YouCook2, respectively.

Methods	Retention Ratio	EventHallusion		VideoComp		VCG-Bench					Performance Drop Rate
		Binary	Desc.	Act.	YC	Consist.	Temp.	Context	Detail	Correct	
Vanilla	100%	63.33	38.74	70.06	70.95	3.32	2.30	3.28	2.38	2.98	-
PruneVid	20%	60.04	32.45	70.58	70.52	3.16	2.15	3.12	2.14	2.86	5.74%
Holitom		63.57	35.76	70.78	70.95	3.22	2.15	3.20	2.22	2.95	2.41%
Holitom+SToP		63.57	38.41	71.04	70.66	3.23	2.18	3.21	2.23	2.96	1.84%$_{\downarrow 0.58\%}$
FastVid		55.99	37.09	60.72	59.81	3.15	2.11	3.16	2.16	2.90	8.21%
FastVid+SToP		62.59	37.75	70.30	71.12	3.15	2.15	3.21	2.24	2.96	2.61%$_{\downarrow 5.60\%}$
VisionZip		58.44	35.76	69.46	69.49	3.22	2.17	3.14	2.14	2.91	4.85%
VisionZip+SToP		62.59	37.42	70.18	70.78	3.16	2.09	3.17	2.19	2.89	3.66%$_{\downarrow 1.19\%}$
PruneVid	15%	58.44	35.10	69.58	70.18	3.14	2.17	3.08	2.11	2.87	5.68%
Holitom		63.08	36.42	69.82	70.01	3.18	2.15	3.17	2.16	2.92	3.52%
Holitom+SToP		64.06	37.42	70.18	70.61	3.15	2.16	3.18	2.20	2.96	2.78%$_{\downarrow 0.74\%}$
FastVid		52.57	30.46	60.00	59.04	3.14	2.10	3.03	2.06	2.80	12.30%
FastVid+SToP		60.88	40.73	69.46	70.61	3.15	2.09	3.14	2.17	2.88	3.42%$_{\downarrow 8.89\%}$
VisionZip		58.44	34.44	67.78	68.29	3.05	2.10	3.03	2.05	2.78	7.87%
VisionZip+SToP		61.61	38.08	68.38	69.84	3.16	2.12	3.12	2.15	2.87	4.36%$_{\downarrow 3.51\%}$
PruneVid	10%	56.72	32.12	68.26	68.38	3.02	2.04	3.02	2.04	2.78	9.22%
Holitom		60.88	36.75	68.38	68.98	3.07	2.06	3.08	2.07	2.84	6.22%
Holitom+SToP		62.59	39.74	69.24	69.10	3.07	2.03	3.10	2.11	2.84	4.80%$_{\downarrow 1.42\%}$
FastVid		49.63	31.46	57.37	58.10	2.95	1.95	2.92	1.94	2.72	15.69%
FastVid+SToP		60.15	36.75	68.38	69.75	3.09	2.03	3.06	2.09	2.83	6.32%$_{\downarrow 9.37\%}$
VisionZip		50.37	28.15	65.39	64.61	2.81	1.84	2.78	1.84	2.55	16.79%
VisionZip+SToP		60.39	36.09	67.78	68.89	3.05	2.06	3.05	2.08	2.80	6.87%$_{\downarrow 9.92\%}$

1) Existing token pruning methods suffer from significant performance drops as the retention ratio decreases. This suggests an inability to preserve fine-grained visual cues under strict token budgets. We attribute this to the fact that when the token budget is tighter, sink tokens with low semantics tend to dominate the selection process, thereby crowding out informative tokens essential for fine-grained video understanding. **2)** Temporal+spatial pruning methods (Holitom, PruneVid) are generally more robust than spatial pruning methods (VisionZip, FastVid), indicating that, as discussed in Sec. 3.3, temporal pruning acts as the primary driver by implicitly suppressing sink tokens. **3)** Applying SToP to existing frameworks (VisionZip, FastVid, Holitom) significantly mitigates performance degradation compared to their original counterparts, especially at a low retention ratio of 10% and 15%. This demonstrates that SToP effectively enhances video understanding by redistributing the selection opportunities toward semantically rich tokens rather than sink tokens with limited semantics.

Results on MCQA benchmarks. Tab. 2 presents a performance comparison across various MCQA benchmarks. From these results, we draw two observations: **1)** Existing token pruning methods show relatively moderate performance drop compared to the sharper declines seen in fine-grained video understanding task. This discrepancy suggests that MCQA benchmarks may not fully capture the limitations of token pruning, highlighting the necessity of fine-grained tasks for rigorous stress-testing. **2)** SToP consistently outperforms baseline counterparts, indicating that by explicitly guiding the pruning of sink tokens, SToP significantly bolsters the model's capacity for video understanding, even in multiple-choice format.

Table 2: Performance comparison on multiple-choice question answering (MCQA) benchmarks. The Performance Drop Rate is computed as in Tab. 1.

Method	Retention	MVBench	VideoMME			NextQA	LongVideoBench	MLVU	Performance Drop Rate
			Short	Medium	Long				
Vanilla	100%	58.54	71.22	56.22	48.61	79.64	54.23	64.66	-
PruneVid	15%	57.26	66.78	54.56	47.22	79.54	54.90	62.65	2.32%
Holitom	15%	57.63	67.89	56.78	48.00	79.74	54.30	62.74	1.31%
Holitom+SToP	15%	57.84	67.89	55.67	48.89	80.38	54.30	63.75	0.95%$_{\downarrow 0.37\%}$
FastVid	15%	57.55	65.89	53.56	48.11	79.24	52.43	62.55	3.15%
FastVid+SToP	15%	57.76	67.67	54.22	49.11	80.30	55.12	63.43	1.18%$_{\downarrow 1.96\%}$
VisionZip	15%	56.42	65.11	54.56	48.11	79.10	52.80	62.65	3.23%
VisionZip+SToP	15%	57.71	67.33	54.89	49.33	79.92	53.63	62.93	1.60%$_{\downarrow 1.63\%}$
PruneVid	10%	56.74	64.89	54.22	47.67	79.10	54.45	61.78	3.17%
Holitom	10%	57.11	64.78	54.44	47.78	79.48	53.55	61.64	3.21%
Holitom+SToP	10%	57.45	66.67	55.22	48.89	79.86	53.33	62.42	2.02%$_{\downarrow 1.20\%}$
FastVid	10%	56.42	62.55	52.67	46.67	78.68	52.58	61.09	5.12%
FastVid+SToP	10%	57.79	65.22	54.22	48.22	79.44	53.55	61.59	2.90%$_{\downarrow 2.22\%}$
VisionZip	10%	54.16	61.00	52.89	45.78	77.40	49.29	60.76	7.36%
VisionZip+SToP	10%	57.34	65.67	53.78	47.67	79.72	52.51	61.96	3.34%$_{\downarrow 4.02\%}$

Results on Cross-Backbone. To further validate the effectiveness of SToP, we extend our experiments to the LLaVA-Video-7B [54] and evaluate performance on EventHallusion and VideoComp. As shown in Tab. 3, we observe that applying SToP to various frameworks (VisionZip, FastVid, Holitom) consistently yields performance gains over their respective baselines. This result aligns with our earlier observations on LLaVA-OneVision, demonstrating the adaptability of SToP.

Table 3: Performance comparison with LLaVA-Video [54].

Method	Retention Ratio	EventHallusion		VideoComp		Performance Drop Rate
		Binary	Desc.	Act.	YC	
Vanilla	100%	69.68	44.37	71.62	70.52	-
PruneVid	15%	62.59	33.11	66.95	68.72	11.62%
Holitom	15%	67.97	44.37	71.62	68.21	1.43%
Holitom+SToP	15%	68.70	45.36	71.26	68.61	0.60%
FastVid	15%	59.17	35.76	57.01	59.55	17.61%
FastVid+SToP	15%	64.79	37.75	66.11	68.81	7.84%
VisionZip	15%	59,41	30.13	67.78	66.92	14.19%
VisionZip+SToP	15%	66.75	42.05	70.42	69.07	3.29%
PruneVid	10%	57.96	31.46	65.51	67.52	14.68%
Holitom	10%	65.77	40.40	69.64	67.87	5.27%
Holitom+SToP	10%	66.50	45.03	70.78	69.07	1.58%
FastVid	10%	55.35	35.10	56.65	59.04	19.66%
FastVid+SToP	10%	62.59	37.42	65.51	67.52	9.66%
VisionZip	10%	44.99	21.85	64.07	64.10	26.46%
VisionZip+SToP	10%	63.57	38.74	69.34	68.04	7.04%

5.3 In-Depth Analysis

Regarding a deeper analysis, we examine the performance of SToP at a retention ratio of 0.1 using the LLaVA-OneVision backbone.

Ablation Studies. In Tab. 4, we conduct ablation studies to understand the effect of each component. To this end, we apply SToP on VisionZip for spatial pruning (S) and Holitom for temporal+spatial pruning (T+S) and evaluate the performance on EventHallusion and VideoComp datasets. We observe that the STSP module significantly improves the performance of VisionZip. It indicates that the STSP module effectively guides the pruning of sink tokens while selecting truly salient tokens. Furthermore, even if Holitom implicitly suppresses sink tokens via temporal pruning, discussed in Sec. 3.3, thereby enhancing video understanding, the STSP modules further improve performance, indicating that there is room for improvement to facilitate the pruning of sink tokens. Finally, incorporating

Table 4: Ablation studies of SToP.

Method	STSP	STTP	EventHallusion		VideoComp		Performance Drop Rate
			Binary	Desc.	Act	YC	
Vanilla			63.33	38.74	70.06	70.95	-
VisionZip (S)			50.37	28.15	65.39	64.61	20.46%
	✓		**60.39**	**36.09**	**67.78**	**68.89**	**4.64%**
Holitom (T+S)			60.88	36.75	68.38	68.98	3.87%
	✓		62.10	37.09	68.02	**69.24**	1.94%
	✓	✓	**62.59**	**39.74**	**69.24**	69.10	**1.17%**

the STTP module into Holitom achieves the best overall performance, demonstrating that the STTP module effectively complements spatial pruning. These results validate the effectiveness of both the STSP and STTP modules.

Different Number of Frames. To investigate how performance varies with the different number of frames (16, 32, 64), we evaluate VisionZip and VisionZip+STOP on the EventHallusion dataset. For each setting, we report the overall performance and the performance drop rate compared to the corresponding vanilla model. As illustrated in Fig. 6, we have the following observations. **1)** The performance of VisionZip linearly increases as the number of frames increases. This suggests that benefit of incorporating additional visual cues outweighs the semantic dilution potentially caused by the accumulation of sink tokens in larger frames. However, this gain comes at the cost of increased inference latency, posing a challenge for real-time applications. **2)** STOP consistently yields substantial gains over the VisionZip baseline across all configurations. Especially, we observe that STOP with only 16 frames surpasses the baseline with 64 frames, highlighting the superiority of STOP in scenarios with a strict token budget. By enhancing the models' fundamental video understanding, STOP allows for high-accuracy inference using significantly fewer frames, thereby optimizing the trade-off between performance and computational efficiency.

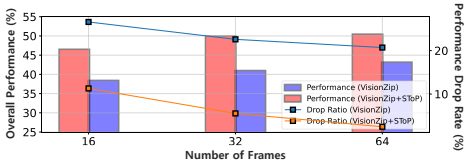

Fig. 6: Performance over the different number of frames.

Comparison with Naive Approaches. To better understand the underlying mechanics of STOP beyond the main results, we compare it against two naive approaches applied to VisionZip on the EventHallusion dataset. For these approaches, we first identify tokens with the top $K\%$ highest attention weights per frame, which often behave as sink tokens (see Sec. 3.2). We then apply two distinct approaches: *Hard Pruning*: these top $K\%$ tokens are deliberately discarded, after which we apply VisionZip, where tokens with high attention are selected. *Attention Redistribution*: Since mitigation of sink tokens was explored by prior work, we attempt to adopt previous work for visual token pruning to see if it is effective. Therefore, following [17], we redistribute the attention weights from these K tokens to the remaining tokens before selection. Fig. 7 illustrates the performance across various K (5% to 20%), from which we draw the following observations: **1)** *Hard Pruning* consistently outperforms the original VisionZip. This confirms our hypothesis that the most attended tokens are often semantically sparse sink tokens. From the result of peaking at $K = 10\%$, we hypothesize that they are likely to stay around tokens with top 10% highest attention. **2)** *Attention Redistribution* underperforms compared to both *Hard pruning* and STOP.

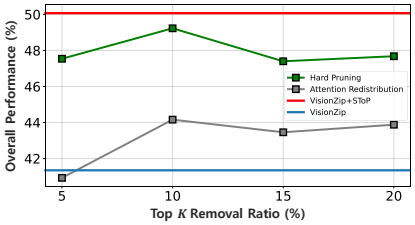

Fig. 7: Performance with naive approaches.

This indicates that visual token pruning requires a different mechanism in that sink-token mitigation methods mainly focus on adjusting inherent information flow, whereas for visual token pruning, explicitly guiding the pruning of sink tokens is necessary. In this vein, *Hard pruning* that directly decides to prune sink tokens is more effective than this approach. **3)** SToP achieves the highest performance, outperforming all naive approaches. While the hard pruning approach risks discarding truly salient tokens that happen to have high attention, SToP combines attention-based information density with a sink score to better distinguish truly salient tokens from sink tokens, leading to more reliable selection of informative tokens.

5.4 Comparison with Other Pruning Approaches

To explore the effectiveness of SToP against other token pruning approach, we compare it against feature-based spatial pruning, a widely adopted alternative that may potentially avoids the sink token issue by not relying on attention scores. Specifically, we employ Density Peaks Clustering with k-Nearest Neighbors (DPC-KNN [9]) algorithm per frame for spatial pruning, following established frameworks in recent work [8, 14, 29, 36]. For the attention-based baseline, we use VisionZip. As shown in Tab. 5, we have the following observations: **1)** At a low retention ratio of 10%, the standard attention-based pruning suffers a significant performance drop. On the other hand, feature-based pruning shows higher performance, as it bypasses the distractions caused by sink tokens. **2)** Applying SToP to the attention-based pruning yields substantial improvements, eventually surpassing the feature-based pruning. This indicates that while naive attention-based pruning is hindered by sink tokens, SToP effectively mitigates this bottleneck, validating its robustness under aggressive spatial pruning.

Table 5: Comparison with feature-based spatial pruning approach.

Method	Retention	EventHallusion		VideoComp		Performance Drop Rate
		Binary	Desc.	Act.	YC	
Vanilla	100%	63.33	38.74	70.06	70.95	-
Feature-based Pruning	15%	57.95	33.44	**68.98**	69.15	6.56%
Attention-based Pruning		58.44	34.44	67.78	68.29	6.46%
+SToP		**61.61**	**38.08**	68.38	**69.84**	**2.10%**
Feature-based Pruning	10%	55.99	35.43	**68.14**	68.72	6.50%
Attention-based Pruning		50.37	28.15	65.39	64.61	15.85%
+SToP		**60.39**	**36.09**	67.78	**68.89**	**4.41%**

6 Related Works

Visual Token pruning in Video LLMs. Recent progress in Video LLMs have introduced visual token pruning to mitigate the high computational overhead associated with processing multiple frames. To achieve this, some methods reduce visual tokens *inside the LLM* to optimize KV cache memory (e.g., FastV [5], DSTP [19]). In contrast, we focus on pruning *before* the visual tokens are fed into the LLM, which maintains full compatibility with optimized attention kernels (e.g., FlashAttention [6]). From this perspective, prior work can be broadly categorized into spatial-only pruning and temporal+spatial pruning approach. For **spatial-only** pruning, VisionZip [48] and FasterVLM [52] utilize vision encoder's attention weights to select salient tokens. FastVid [36] adopts a similar attention-based selection strategy, and further applies density-based token merging to the remaining tokens. CDPruner [53] reframes pruning through a detrimental point

process to enable dynamic token pruning. To further exploit redundancy across frames, several methods perform **temporal+spatial pruning**. PruneVid [14] and HoliTom [35] segment videos into clips to first reduce temporal redundancy via inter-frame similarity, followed by spatial clustering (e.g., DPC-KNN [9]) or attention-based pruning. More recently, STTM [15] and FlashVid [10] have introduced tree-based structures for hierarchical spatio-temporal reduction. In contrast, Dycoke [41] performs only temporal pruning, which limits the achievable token reduction (up to 25%). Despite their efficiency, existing works are predominantly evaluated on coarse-grained video understanding tasks, such as MCQA. On the other hand, we shifts the focus toward fine-grained video understanding (e.g., hallucination or compositional reasoning), where we observe that sink token is a key obstacle, especially at low token retention rates.

Attention Sink. The attention sink phenomenon in the transformer-based models refers to the tendency for a small subset of low-semantic tokens to attract a disproportionate amount of attention. This phenomenon was first documented in Large Language Models (LLMs), where certain tokens with minimal semantic meaning (e.g., BOS, SEP token) consistently receive extremely high attention scores [40, 45]. Similar to the sinks found in LLMs, several works [7, 11, 16, 21, 28, 39] have identified an analogous phenomenon in Vision Transformers (ViTs), where the small subset of visual tokens with low semantics receive excessive attention. While prior research has sought to mitigate these sinks to improve ViT interpretability of ViTs [7, 16] or strengthen internal information flow [11, 28, 39], their impact on visual token pruning remains underexplored. In this work, we demonstrate that these sink tokens are a key obstacle to effective visual token pruning in fine-grained video tasks, motivating us to propose STOP.

7 Conclusion

In this work, we revisit a key assumption in visual token pruning: whether existing methods indeed preserve fine-grained visual cues. We argue that this has been under-examined since most prior work is validated mainly on MCQA benchmarks, where coarse-grained cues are often sufficient. From our analysis, we identify sink tokens as a key bottleneck for fine-grained video understanding. Building on this insight, we propose STOP, which introduces a sink score to quantify each token's tendency to behave as a sink. We then use this score within established spatial and temporal pruning frameworks, implemented via the STSP and STTP modules, respectively. Extensive experiments show that STOP consistently strengthens fine-grained video understanding under aggressive pruning, delivering clear performance gains, especially at a 10% retention ratio. Moreover, STOP makes few-frame inference substantially more effective, outperforming the baseline that requires many more frames to achieve comparable understanding. Regarding limitations and future work, please refer to Appendix H.

More broadly, we believe this work offers two key contributions to the visual token pruning community: 1) *Spatial Insight*: explicitly addressing sink tokens is crucial for capturing the fine-grained visual cues in attention-based pruning. 2) *Temporal Insight*: temporal pruning can serve as a potent sink token suppressor, extending its utility beyond mere temporal redundancy reduction.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2020-0-00004), National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS-2024-00406985), and National Research Foundation of Korea(NRF) funded by Ministry of Science and ICT (RS-2022-NR068758).

References

1. Alvar, S.R., Singh, G., Akbari, M., Zhang, Y.: Divprune: Diversity-based visual token pruning for large multimodal models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 9392–9401 (2025)
2. Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al.: Qwen technical report. arXiv preprint arXiv:2309.16609 (2023)
3. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report (2025), https://arxiv.org/abs/2502.13923
4. Chen, G., Liu, Y., Huang, Y., He, Y., Pei, B., Xu, J., Wang, Y., Lu, T., Wang, L.: Cg-bench: Clue-grounded question answering benchmark for long video understanding. arXiv preprint arXiv:2412.12075 (2024)
5. Chen, L., Zhao, H., Liu, T., Bai, S., Lin, J., Zhou, C., Chang, B.: An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In: European Conference on Computer Vision. pp. 19–35. Springer (2024)
6. Dao, T., Fu, D., Ermon, S., Rudra, A., Ré, C.: Flashattention: Fast and memory-efficient exact attention with io-awareness. Advances in neural information processing systems **35**, 16344–16359 (2022)
7. Darcet, T., Oquab, M., Mairal, J., Bojanowski, P.: Vision transformers need registers. arXiv preprint arXiv:2309.16588 (2023)
8. Dhouib, M., Buscaldi, D., Vanier, S., Shabou, A.: Pact: Pruning and clustering-based token reduction for faster visual language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 14582–14592 (2025)
9. Du, M., Ding, S., Jia, H.: Study on density peaks clustering based on k-nearest neighbors and principal component analysis. Knowledge-Based Systems **99**, 135–145 (2016)
10. Fan, Z., Chen, K., Xing, R., Li, Y., Jiang, L., Tian, Z.: Flashvid: Efficient video large language models via training-free tree-based spatiotemporal token merging. arXiv preprint arXiv:2602.08024 (2026)
11. Feng, W., Wang, H., Wang, J., Zhang, X., Zhao, J., Liang, Y., Chen, X., Han, D.: Edit: enhancing vision transformers by mitigating attention sink through an encoder-decoder architecture. In: International Conference on Optoelectronics, Computer Science, and Algorithms (OCSA 2025). vol. 14008, pp. 246–259. SPIE (2026)
12. Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., et al.: Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24108–24118 (2025)

13. Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al.: The llama 3 herd of models. arXiv preprint arXiv:2407.21783 (2024)
14. Huang, X., Zhou, H., Han, K.: Prunevid: Visual token pruning for efficient video large language models. In: Findings of the Association for Computational Linguistics: ACL 2025. pp. 19959–19973 (2025)
15. Hyun, J., Hwang, S., Han, S.H., Kim, T., Lee, I., Wee, D., Lee, J.Y., Kim, S.J., Shim, M.: Multi-granular spatio-temporal token merging for training-free acceleration of video llms. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23990–24000 (2025)
16. Jiang, N., Dravid, A., Efros, A., Gandelsman, Y.: Vision transformers don't need trained registers. arXiv preprint arXiv:2506.08010 (2025)
17. Kang, S., Kim, J., Kim, J., Hwang, S.J.: See what you are told: Visual attention sink in large multimodal models. arXiv preprint arXiv:2503.03321 (2025)
18. Kim, D., Piergiovanni, A., Mallya, G., Angelova, A.: Videocomp: Advancing fine-grained compositional and temporal alignment in video-text models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 29060–29070 (2025)
19. Kim, J., Kim, K., Kim, W., Lee, B.K., Park, C.: Why and when visual token pruning fails? a study on relevant visual information shift in mllms decoding. arXiv preprint arXiv:2604.12358 (2026)
20. Kim, J., Kim, K., Seo, S., Park, C.: Compodistill: Attention distillation for compositional reasoning in multimodal llms. arXiv preprint arXiv:2510.12184 (2025)
21. Kim, S., Kim, J., Yeom, T., Park, W., Kim, K., Lee, J.: Activation quantization of vision encoders needs prefixing registers. arXiv preprint arXiv:2510.04547 (2025)
22. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024)
23. Li, K., Wang, Y., He, Y., Li, Y., Wang, Y., Liu, Y., Wang, Z., Xu, J., Chen, G., Luo, P., et al.: Mvbench: A comprehensive multi-modal video understanding benchmark. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22195–22206 (2024)
24. Li, Y., Wang, C., Jia, J.: Llama-vid: An image is worth 2 tokens in large language models. In: European Conference on Computer Vision. pp. 323–340. Springer (2024)
25. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. In: Proceedings of the 2024 conference on empirical methods in natural language processing. pp. 5971–5984 (2024)
26. Lin, J., Yin, H., Ping, W., Molchanov, P., Shoeybi, M., Han, S.: Vila: On pretraining for visual language models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 26689–26699 (2024)
27. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In: European conference on computer vision. pp. 38–55. Springer (2024)
28. Lu, A., Liao, W., Wang, L., Yang, H., Shi, J.: Artifacts and attention sinks: Structured approximations for efficient vision transformers. arXiv preprint arXiv:2507.16018 (2025)
29. Ma, J., Zhang, Q., Lu, M., Wang, Z., Zhou, Q., Song, J., Zhang, S.: Mmg-vid: Maximizing marginal gains at segment-level and token-level for efficient video llms. arXiv preprint arXiv:2508.21044 (2025)

30. Maaz, M., Rasheed, H., Khan, S., Khan, F.: Video-chatgpt: Towards detailed video understanding via large vision and language models. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 12585–12602 (2024)
31. Mangalam, K., Akshulakov, R., Malik, J.: Egoschema: A diagnostic benchmark for very long-form video language understanding. Advances in Neural Information Processing Systems **36**, 46212–46244 (2023)
32. Qiu, H., Gao, M., Qian, L., Pan, K., Yu, Q., Li, J., Wang, W., Tang, S., Zhuang, Y., Chua, T.S.: Step: Enhancing video-llms' compositional reasoning by spatio-temporal graph-guided self-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3284–3294 (2025)
33. Rawal, R., Shirkavand, R., Huang, H., Somepalli, G., Goldstein, T.: Argus: Hallucination and omission evaluation in video-llms. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20280–20290 (2025)
34. Shang, Y., Cai, M., Xu, B., Lee, Y.J., Yan, Y.: Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22857–22867 (2025)
35. Shao, K., Tao, K., Qin, C., You, H., Sui, Y., Wang, H.: Holitom: Holistic token merging for fast video large language models. arXiv preprint arXiv:2505.21334 (2025)
36. Shen, L., Gong, G., He, T., Zhang, Y., Liu, P., Zhao, S., Ding, G.: Fastvid: Dynamic density pruning for fast video large language models. arXiv preprint arXiv:2503.11187 (2025)
37. Shen, X., Xiong, Y., Zhao, C., Wu, L., Chen, J., Zhu, C., Liu, Z., Xiao, F., Varadarajan, B., Bordes, F., et al.: Longvu: Spatiotemporal adaptive compression for long video-language understanding. arXiv preprint arXiv:2410.17434 (2024)
38. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., et al.: Openai gpt-5 system card. arXiv preprint arXiv:2601.03267 (2025)
39. Sinks, E.A.V., Sinks, D.L.e., Sinks, C.P.V., Sinks, B., Image, A.G.: To sink or not to sink: Visual information pathways in large vision-language models
40. Sun, M., Chen, X., Kolter, J.Z., Liu, Z.: Massive activations in large language models. arXiv preprint arXiv:2402.17762 (2024)
41. Tao, K., Qin, C., You, H., Sui, Y., Wang, H.: Dycoke: Dynamic compression of tokens for fast video large language models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 18992–19001 (2025)
42. Tatler, B.W.: The central fixation bias in scene viewing: Selecting an optimal viewing position independently of motor biases and image feature distributions. Journal of vision **7**(14), 4–4 (2007)
43. Wang, A., Sun, F., Chen, H., Lin, Z., Han, J., Ding, G.: [cls] token tells everything needed for training-free efficient mllms. arXiv preprint arXiv:2412.05819 (2024)
44. Wu, H., Li, D., Chen, B., Li, J.: Longvideobench: A benchmark for long-context interleaved video-language understanding, 2024. URL https://arxiv. org/abs/2407 **15754**(8)
45. Xiao, G., Tian, Y., Chen, B., Han, S., Lewis, M.: Efficient streaming language models with attention sinks. arXiv preprint arXiv:2309.17453 (2023)
46. Xiao, J., Shang, X., Yao, A., Chua, T.S.: Next-qa: Next phase of question-answering to explaining temporal actions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9777–9786 (2021)
47. Xu, L., Zhao, Y., Zhou, D., Lin, Z., Ng, S.K., Feng, J.: Pllava: Parameter-free llava extension from images to videos for video dense captioning. arXiv preprint arXiv:2404.16994 (2024)

48. Yang, S., Chen, Y., Tian, Z., Wang, C., Li, J., Yu, B., Jia, J.: Visionzip: Longer is better but not necessary in vision language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19792–19802 (2025)
49. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11975–11986 (2023)
50. Zhang, B., Li, K., Cheng, Z., Hu, Z., Yuan, Y., Chen, G., Leng, S., Jiang, Y., Zhang, H., Li, X., et al.: Videollama 3: Frontier multimodal foundation models for image and video understanding. arXiv preprint arXiv:2501.13106 (2025)
51. Zhang, J., Jiao, Y., Chen, S., Zhao, N., Tan, Z., Li, H., Ma, X., Chen, J.: Eventhallusion: Diagnosing event hallucinations in video llms. arXiv preprint arXiv:2409.16597 (2024)
52. Zhang, Q., Cheng, A., Lu, M., Zhuo, Z., Wang, M., Cao, J., Guo, S., She, Q., Zhang, S.: [cls] attention is all you need for training-free visual token pruning: Make vlm inference faster. arXiv e-prints pp. arXiv–2412 (2024)
53. Zhang, Q., Liu, M., Li, L., Lu, M., Zhang, Y., Pan, J., She, Q., Zhang, S.: Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. arXiv preprint arXiv:2506.10967 (2025)
54. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: Llava-video: Video instruction tuning with synthetic data. arXiv preprint arXiv:2410.02713 (2024)
55. Zhou, J., Shu, Y., Zhao, B., Wu, B., Liang, Z., Xiao, S., Qin, M., Yang, X., Xiong, Y., Zhang, B., et al.: Mlvu: Benchmarking multi-task long video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13691–13701 (2025)