Unified Spatiotemporal Token Compression for Video-LLMs at Ultra-Low Retention

Supplementary Material

7. Preliminaries

The inference process of Video-LLMs typically involves three phases: *encoding*, *prefilling*, and *decoding*.

(1) Encoding Phase. Given an input video containing B frames, the visual encoder first processes each frame individually to generate N_v visual embedding vectors. A projector then maps these visual embeddings into the text embedding space, producing the visual tokens $\boldsymbol{H}_v \in \mathbb{R}^{BN_v \times d}$, where d denotes the dimension of the LLM hidden state. Simultaneously, the text prompt $T = \{t_i\}_{i=1}^{N_q}$ is tokenized and passed through an embedding layer to obtain the token embeddings $\boldsymbol{H}_q \in \mathbb{R}^{N_q \times d}$. Finally, the visual tokens $\boldsymbol{H}_v$ and text token embeddings $\boldsymbol{H}_q$ are concatenated into a unified sequence $\boldsymbol{H} = \text{concat}[\boldsymbol{H}_v, \boldsymbol{H}_q]$, which serves as the input to the LLM.

(2) Prefilling Phase. During the prefilling phase, each Transformer layer l in the LLM conducts self-attention over the input $\boldsymbol{H}$. Specifically, each layer first projects $\boldsymbol{H}$ into query $\boldsymbol{Q}^l$, key $\boldsymbol{K}^l$, and value $\boldsymbol{V}^l$ matrices, calculated according to the following formula:

$$\boldsymbol{Q}^l = \boldsymbol{H}\boldsymbol{W}_Q^l, \quad \boldsymbol{K}^l = \boldsymbol{H}\boldsymbol{W}_K^l, \quad \boldsymbol{V}^l = \boldsymbol{H}\boldsymbol{W}_V^l, \quad \text{(S1)}$$

where $\boldsymbol{W}_Q^l, \boldsymbol{W}_K^l, \boldsymbol{W}_V^l \in \mathbb{R}^{d \times d}$ are learnable projection matrices. The resulting $\boldsymbol{K}^l$ and $\boldsymbol{V}^l$ matrices are then stored in the KV cache to expedite computation during the subsequent decoding phase.

(3) Decoding Phase. During the decoding phase, the model generates output tokens autoregressively while dynamically accessing and updating the KV cache. At each time step t, the LLM computes the query, key, and value solely from the latest generated token h_t. The resulting $\boldsymbol{K}$ and $\boldsymbol{V}$ are then appended incrementally to the KV cache:

$$\boldsymbol{K} = [\boldsymbol{K}, h_t\boldsymbol{W}_K], \quad \boldsymbol{V} = [\boldsymbol{V}, h_t\boldsymbol{W}_V], \quad \text{(S2)}$$

This process eliminates redundant attention computations over previously processed tokens and substantially enhances decoding efficiency.

8. BenchMarks

We evaluate our method on various video understanding benchmarks, detailed as follows:

- **MVBench** [13] formulates 20 video understanding tasks, each with 200 QA pairs, to assess temporal comprehension beyond single-frame analysis and provide a comprehensive model evaluation.
- **EgoSchema** [19] consists of over 5000 human-curated multiple-choice question answer pairs, spanning over 250 hours of real video data, covering a very broad range of natural human activity and behavior. For each question, EgoSchema requires the correct answer to be selected between five given options based on a three-minute-long video clip.
- **MLVU** [48] designed for long-form videos, features a benchmark with durations from 3 minutes to over 2 hours (12-minute average). It covers diverse genres like movies, documentaries, and TV series, while evaluating models across 9 distinct tasks such as topic reasoning, video summarization, and needle question answering.
- **LongVideoBench** [34] consists of 3,763 videos and 6,678 associated multiple-choice questions, covering diverse domains such as movies and news, specifically designed to assess a model's capacity for temporal information retrieval and analysis.
- **VideoMME** [8] consists of 900 videos and 2,700 QA pairs with durations from 11 seconds to 1 hour. These are categorized into three temporal subsets (short-, medium-, and long-term) and span 6 main visual domains, such as life record and knowledge.
- **ActivityNet-QA** [40] contains 58,000 human-annotated QA pairs on 5,800 videos derived from the popular ActivityNet dataset. It provides a benchmark for testing the performance of VideoQA models on long-term spatiotemporal reasoning. Moreover, LLMs are employed to assess the quality of model-generated responses, providing a more flexible and nuanced evaluation compared to traditional accuracy-based metrics.

9. Computing Cost Estimation.

To quantitatively assess the computational efficiency improvement achieved by token compression, we adopt floating-point operations (FLOPs) during the prefilling and decoding phases as the evaluation metric, following prior work [4, 23, 28, 35]. The total FLOPs of the Transformer model are formulated as:

$$\text{FLOPs} = \sum_{i=1}^{T}((4n_i d^2 + 2n_i^2 d + 2n_i dm) + R((4d^2 + 2dm) + 2(dn_i + \frac{d}{2}(R+1))), \quad \text{(S3)}$$

where T denotes the total number of layers, n_i indicates the number of input tokens after pruning at layer i, d is the

Table S1. Comparison results based on the Qwen2.5-VL-7B model.

Method	Retention Ratio	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. ↑ Score	Avg. ↑ %
Qwen2.5-VL [2]	100%	63.0	52.8	38.5	55.0	57.5	53.4	100.0
VisionZip [38]	2%	50.3	45.3	29.7	43.4	44.1	42.6	79.8
FastVID [25]	2%	51.8	46.2	30.6	45.0	46.4	44.0	82.5
Ours(w/o M)	2%	**54.1**	47.8	**32.7**	45.4	**48.1**	**45.6**	**85.5**
Ours	2%	53.7	**48.4**	31.8	**45.6**	**48.1**	45.5	85.3
VisionZip [38]	1%	46.1	42.3	25.8	41.8	41.4	39.5	74.0
FastVID [25]	1%	47.2	43.5	26.4	**45.1**	43.1	41.1	76.9
Ours(w/o M)	1%	50.9	45.0	28.3	43.2	44.4	42.4	79.4
Ours	1%	**51.0**	**45.5**	**28.6**	43.4	**44.5**	**42.6**	**79.8**

hidden dimension, m is the intermediate size of the feedforward network (FFN), and $R = 100$ is the fixed number of tokens generated during decoding.

10. Additional Results

To provide a more comprehensive evaluation, we further include results at a higher retention ratio (15%) and reproduce DyToK [16] under the same setting (Tab. S2). The results show that our method remains state-of-the-art across all benchmarks. Notably, at 15% retention, our performance nearly matches that of the full model. Meanwhile, our advantage is more pronounced in low-retention regimes, where efficient token utilization becomes critical. As the retention ratio increases, all methods gradually converge to a similar performance plateau.

11. Results on Qwen2.5-VL-7B

To evaluate the generality of our proposed method across different model architectures, we conduct additional experiments using the structurally distinct foundation model Qwen2.5-VL-7B [2]. This model incorporates a vision encoder based on window attention, which enables dynamic adjustment of the input video frame resolution. As shown in Tab. S1, our approach maintains strong performance on this architecture, achieving 85.5% of the original model's performance with only 2% of total tokens, while substantially surpassing existing methods. These results further confirm the robust generalization capability of our method across diverse vision-language architectures.

12. Result on ActivityNet-QA

We also evaluate the proposed method on the ActivityNet [40], an open-ended question-answering benchmark. This task requires free-form answer generation, unlike multiple-choice video QA. Using GPT-3.5-Turbo as an evaluator, our approach is shown to outperform comparable methods in both accuracy and generative quality,as described in Tab. S3.

13. Higher Frame Sampling Rates

Our approach demonstrates robust performance advantages over Holitom at high frame sampling rates across key long-video benchmarks, including MLVU [48], LongVideoBench [34], and VideoMME [8], as detailed in Tab. S4. This consistent outperformance underscores its superior effectiveness for long-duration video understanding at ultra-low token retention.

14. More Ablations

Ablation on pruning layer K within LLM. Ablation study in Tab. S5 reveals that the effectiveness of the Text-Aware pruning strategy depends on its integration depth within the LLM. In shallow layers, performance is limited by insufficient alignment between visual features and textual semantics. In deeper layers, performance declines due to interference from unfiltered irrelevant visual information. The strategy performs optimally in intermediate layers, where visual-semantic integration is well-established and redundancy remains controllable, thereby enhancing the refinement of key information and improving task accuracy.

Ablation on pruning ratio R within LLM. As illustrated in Fig. S1, different pruning ratios have distinct effects across network layers. While aggressive pruning in early layers substantially reduces FLOPs, it leads to sharp performance degradation. In contrast, deeper layers exhibit more aggregated visual features and increased token redundancy, making them more tolerant to higher pruning ratios. An effective balance between accuracy and computation therefore requires selecting appropriate pruning ratios according to layer depth.

Table S2. Additional Comparison with DyToK under Higher Retention Ratios

Method	Retention Ratio	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. ↑ Score	Avg. ↑ %
LLaVA-OV-7B [11]	100%	58.3	60.4	47.7	56.4	58.6	56.3	100
VisionZip [38]	15%	56.5	59.8	43.3	54.4	56.1	54.0	95.9
DyToK [16]	15%	56.1	59.5	44.6	53.7	56.1	54.0	95.9
FastVID [25]	15%	56.0	58.8	43.3	56.2	**57.7**	54.4	96.6
HoliTom [23]	15%/7.5%	58.1	**61.2**	**46.7**	56.4	57.3	56.0	99.5
Ours	15%/7.5%	**58.3**	60.5	46.5	**57.5**	**57.7**	**56.1**	**99.6**
FsatV [4]	100%/10%	53.2	55.9	41.6	52.1	52.7	51.1	90.8
VisionZip [38]	10%	53.5	58.0	42.5	49.3	53.4	51.3	91.2
LLaVA-Scissor [27]	10%	-	57.5	-	-	55.8	-	-
DyToK [16]	10%	55.3	58.4	42.4	52.3	54.7	52.6	93.5
FastVID [25]	10%	55.9	58.7	42.6	56.3	**57.3**	54.2	96.2
HoliTom [23]	10%/5%	57.3	**61.2**	45.1	56.3	56.8	55.3	98.3
Ours (w/o M)	10%	57.4	60.5	44.7	**57.4**	56.6	55.3	98.3
Ours	10%/5%	**57.7**	60.6	**45.5**	56.4	56.6	**55.4**	**98.4**
FastV [4]	100%/5%	51.2	53.9	35.8	47.9	49.7	47.7	84.7
VisionZip [38]	5%	45.2	51.9	37.4	46.4	48.2	45.8	81.4
LLaVA-Scissor [27]	5%	-	56.6	-	-	53.3	-	-
DyToK [16]	5%	50.0	53.9	40.8	49.1	50.6	48.9	86.8
FastVID [25]	5%	53.0	57.1	42.1	51.1	54.2	51.5	91.5
HoliTom [23]	5%/2.5%	55.6	**60.5**	40.6	53.6	54.2	52.9	94.0
Ours (w/o M)	5%	**56.4**	59.5	42.5	53.7	**55.0**	53.4	94.9
Ours	5%/2.5%	**56.4**	60.2	**42.8**	**54.5**	54.7	**53.7**	**95.4**
FastV [4]	100%/2%	49.0	50.6	34.1	47.1	47.3	45.6	81.0
VisionZip [38]	2%	41.7	47.6	31.8	45.1	45.9	42.4	75.3
DyToK [16]	2%	42.8	48.4	33.3	45.9	47.1	43.5	77.3
FastVID [25]	2%	48.0	52.3	37.6	47.3	49.2	46.9	83.3
HoliTom [23]	2%/1%	52.6	57.2	37.4	48.5	51.1	49.4	87.7
Ours (w/o M)	2%	**52.9**	57.2	39.5	**51.0**	51.3	50.4	89.5
Ours	2%/1%	52.8	**57.6**	**40.3**	50.8	**51.8**	**50.7**	**90.1**
FastV [4]	100%/1%	48.2	48.8	32.3	45.5	46.2	44.2	78.5
VisionZip [38]	1%	40.8	43.8	29.7	44.4	44.3	40.6	72.1
DyToK [16]	1%	40.7	44.3	30.7	45.6	45.0	41.3	73.3
FastVID [25]	1%	45.3	47.8	32.4	46.1	47.0	43.7	77.7
HoliTom [23]	1%/0.5%	49.6	52.9	33.9	48.1	49.0	46.7	82.9
Ours (w/o M)	1%	**50.5**	53.3	**34.4**	**49.1**	**49.8**	**47.4**	**84.2**
Ours	1%/0.5%	**50.5**	**53.8**	**34.4**	48.8	49.2	47.3	84.1

15. Detailed Hyperparameter Analysis

All experiments are conducted under a unified configuration described in Section 4.1. We provide a comprehensive analysis of key hyperparameters, including τ, λ, and the clustering ratio, with results summarized in Tabs. S6 to S8. The selected values are chosen based on their overall performance across different settings, ensuring a stable and robust configuration. These results further validate the sensitivity and effectiveness of each component in our method.

16. Visualizations

Unified Spatiotemporal Compression. The unified spatiotemporal compression process is visualized in Fig. S3. Red tokens indicate pruned redundant tokens transferred to the recycling pool through similarity-based pruning, whereas green tokens represent preserved key information. This process effectively eliminates spatial redundancy. Moreover, clustering tokens from the recycling pool yields purple tokens that serve as semantic supplements, maintain-

Table S3. Comparison on ActivityNet-QA. The two metrics (*Accuracy* and *Score*) are sub-scores under the ActivityNet-QA benchmark.

Method	Retain Ratio	ActivityNet-QA	
		Accuracy ↑	Score ↑
LLaVA-OV-7B	100%	54.5	3.55
+ VisionZip [38]	2%	41.7	3.13
+ FastVid [25]	2%	49.6	3.35
+ HoliTom [23]	2%	49.9	3.31
+ Ours	2%	**50.2**	**3.37**

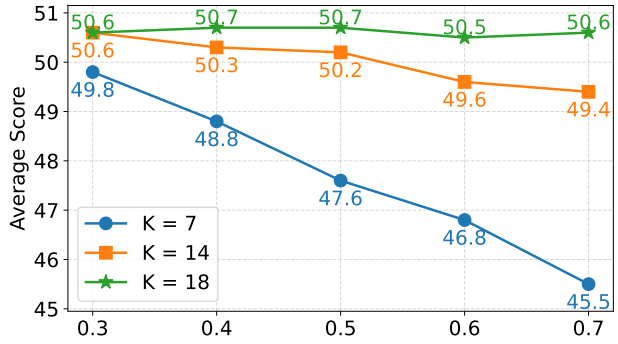


Figure S1. Performance of different pruning ratios across different layers

ing full semantic integrity of the video content.

Text-aware Merging. Figure S2 illustrates the behavior of the text-aware mechanism in the LLM. For identical video inputs, our approach dynamically attends to question-relevant visual tokens while filtering out irrelevant information. This targeted filtering mechanism enhances the precision of the generated responses.

17. Limitations and Future Work

Despite its effectiveness in accelerating video LLM inference under ultra-low retention ratios, our approach has certain limitations. It currently supports only fixed offline videos and lacks real-time streaming token compression. Furthermore, uniform frame sampling can result in redundant frames for short videos while potentially missing critical segments in long videos. Future work will focus on jointly optimizing frame selection and token compression to enhance performance in extreme low-retention settings.

Table S4. Comparison across different frame settings at 2% token retention.

Method	Frames	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. Score
Vanilla [11]	16	58.1	59.5	41.8	55.7	56.7	54.4
HoliTom [23]	64	54.8	59.5	38.6	51.0	54.4	51.7
Ours	64	54.1	59.1	41.3	52.5	54.5	52.3
HoliTom [23]	96	55.9	60.4	41.0	51.4	54.1	52.6
Ours	96	55.7	59.5	42.9	54.2	55.9	53.6
HoliTom [23]	128	55.9	**60.9**	41.3	53.8	55.4	53.5
Ours	128	**56.1**	59.5	**47.9**	**55.2**	**58.0**	**55.3**

Table S5. Performance comparison of different pruning layer K.

Method	K	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. Score	Avg. %
Vanilla [11]	-	58.3	60.4	47.7	56.4	58.6	56.3	100
Ours	2	48.7	52.7	36.7	47.8	49.4	47.1	83.6
Ours	7	50.3	54.0	36.6	47.8	49.5	47.6	84.6
Ours	14	**52.9**	55.7	39.7	**51.1**	51.4	50.2	89.1
Ours	18	52.8	**57.6**	**40.3**	50.8	**51.8**	**50.7**	**90.1**
Ours	21	52.8	57.1	39.9	50.3	51.7	50.4	89.5

Table S6. Comparison of τ.

Method	τ	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. Score	Avg. %
Vanilla [11]	-	58.3	60.4	47.7	56.4	58.6	56.3	100
Ours	0.5	49.6	52.9	39.4	49.1	52.3	48.7	86.4
Ours	0.6	51.7	55.9	**40.0**	50.7	**52.7**	50.3	89.4
Ours	0.7	52.9	**57.2**	39.5	**51.0**	51.3	**50.4**	**89.5**
Ours	0.8	**53.1**	56.4	38.2	49.4	51.3	49.7	88.2
Ours	0.9	52.1	52.4	37.1	49.7	50.7	48.4	86.0

Table S7. Comparison of Cluster Ratio at 2% Tokens.

Method	Ratio	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. Score	Avg. %
Vanilla	-	58.3	60.4	47.7	56.4	58.6	56.3	100
Ours	0.1	51.1	56.5	39.0	50.7	51.6	49.7	88.4
Ours	0.2	51.6	56.8	38.6	49.9	51.3	49.6	88.2
Ours	0.3	**52.9**	**57.2**	**39.5**	51.0	51.3	**50.4**	**89.5**
Ours	0.4	52.2	56.5	38.4	49.4	**52.0**	49.7	88.3
Ours	0.5	52.4	56.5	38.7	**51.2**	**52.0**	50.2	89.1
Ours	0.6	52.8	57.0	37.6	51.0	51.2	49.9	88.7
Ours	0.7	52.6	56.1	36.1	50.5	51.3	49.3	87.6

Table S8. Comparison of λ.

Method	λ	MVBench	EgoSchema	MLVU	LongVideo Bench	VideoMME	Avg. Score	Avg. %
Vanilla [11]	-	58.3	60.4	47.7	56.4	58.6	56.3	100
Ours	0	52.6	57.5	**40.4**	50.6	51.5	50.5	89.7
Ours	0.25	52.6	57.5	40.1	50.7	51.5	50.5	89.7
Ours	0.5	**52.8**	**57.6**	40.3	**50.8**	**51.8**	**50.7**	**90.1**
Ours	0.75	52.5	57.5	39.9	50.5	51.5	50.4	89.5
Ours	1	52.6	**57.6**	40.0	50.6	51.6	50.5	89.7

Figure S2. Tokens selected by our text-aware module within the LLM.

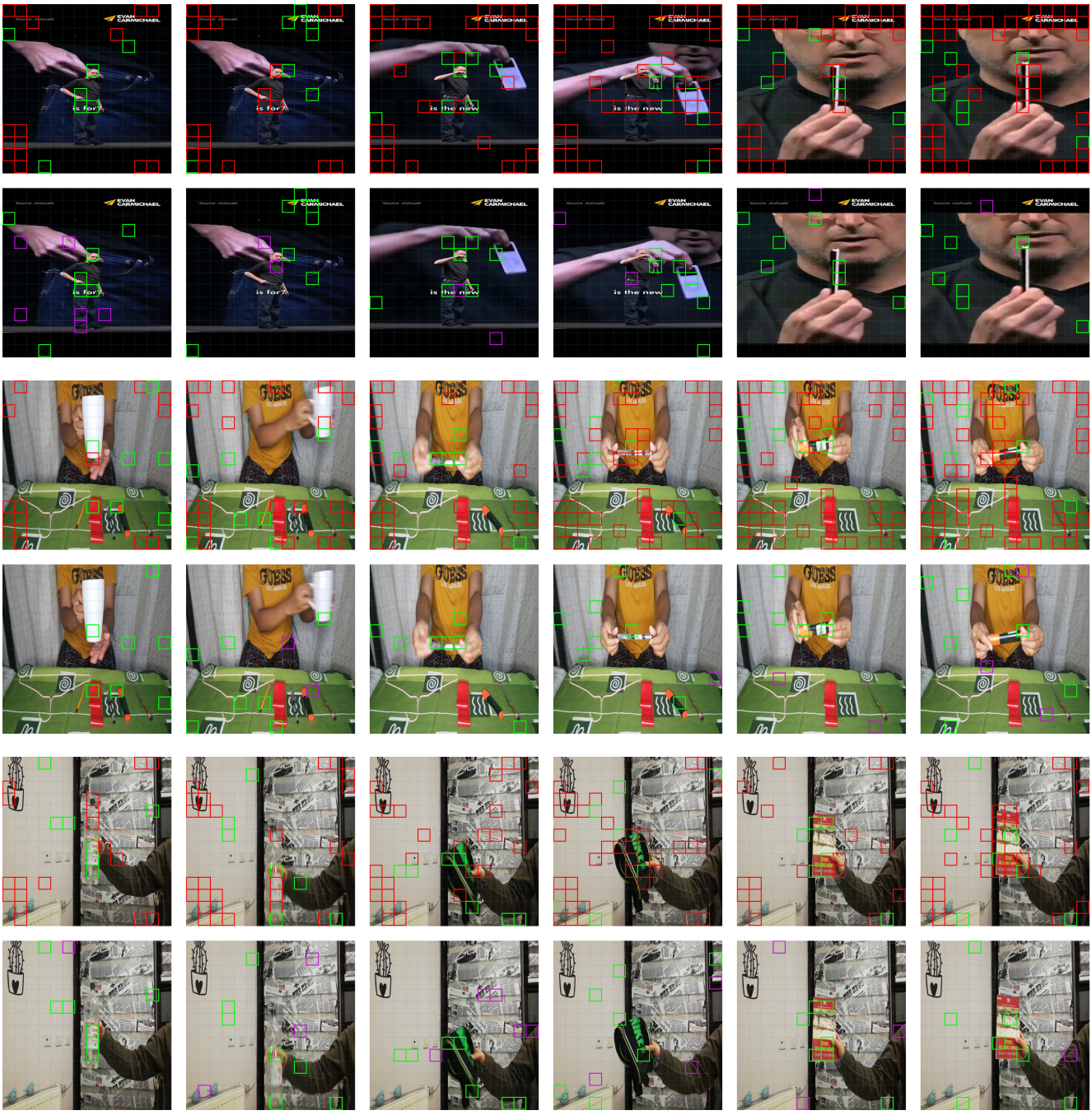
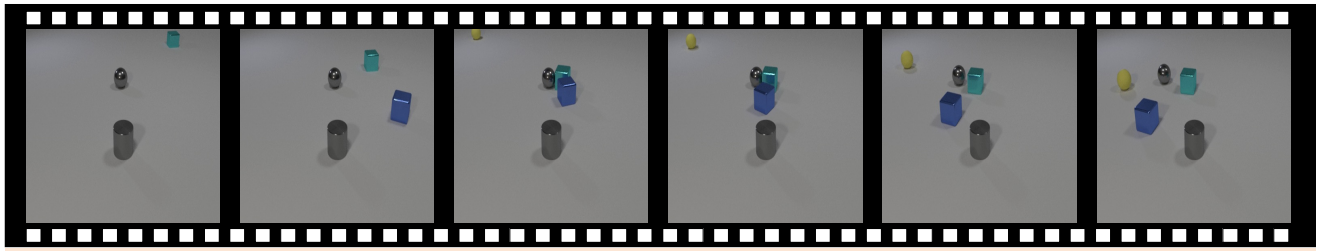


Figure S3. Token reductions outside the LLM. Green boxes mark retained tokens, Red boxes mark tokens discarded by similarity filtering, and Purple boxes mark tokens merged via clustering.



: What color is the object that is stationary? Correct Incorrect

LLaVA-OV-7B:	The object that is stationary is blue.
VisionZip:	The object that is stationary is yellow.
HoliTom:	Blue.
Ours:	Gray.

: What activity does the video depict? Correct Incorrect

LLaVA-OV-7B:	The video depicts a person holding and examining a coin, then moving their hand away from the camera to reveal more of the room.
VisionZip:	The video depicts a person's hand opening a refrigerator door.
HoliTom:	The video depicts a person's hand performing a series of movements, possibly as part of a game or exercise.
Ours:	The video depicts a person's hand holding a coin and then dropping it.

Figure S4. Qualitative generation comparison between our method and other approaches.