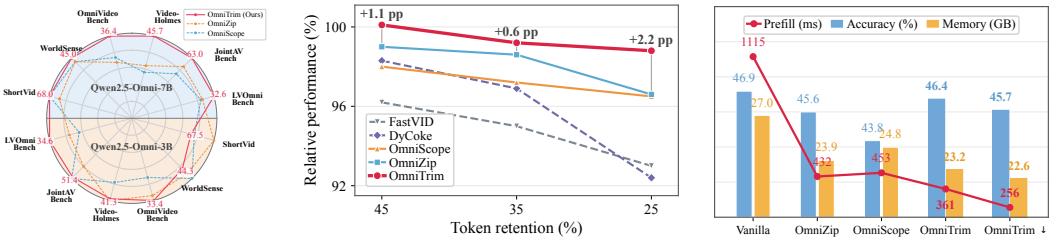
DYNAMIC AUDIO-VISUAL CORRELATION-AWARE TOKEN COMPRESSION FOR OMNIMODAL LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review



(a) Benchmark results (b) Performance vs. retention (c) Inference efficiency

Figure 1: **OmniTrim achieves state-of-the-art performance with efficient inference.** (a) OmniTrim achieves the highest average performance across six audio-visual benchmarks. (b) OmniTrim remains ahead as compression increases, with less performance degradation. (c) OmniTrim achieves higher accuracy than the baselines with faster prefill and lower peak GPU memory.

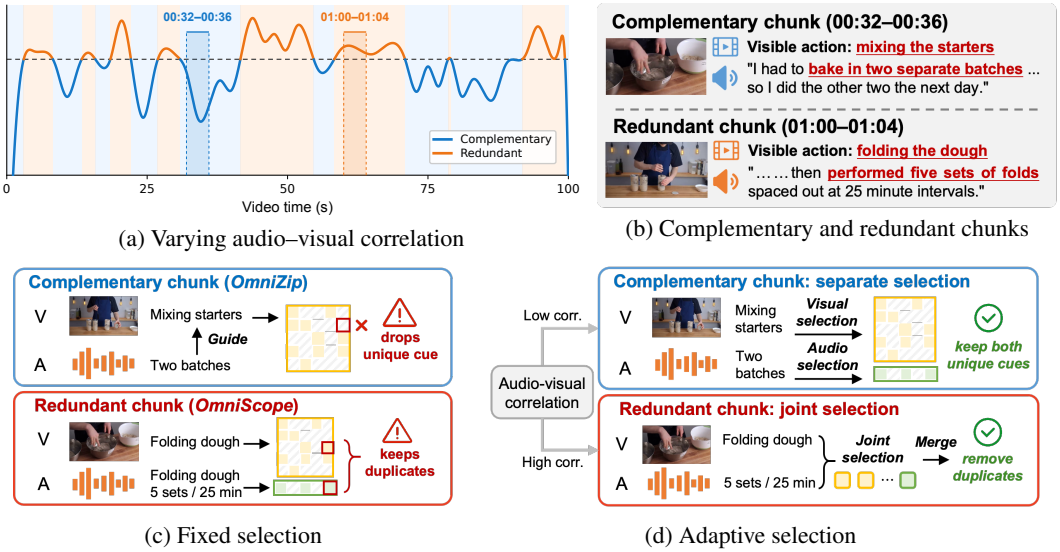
ABSTRACT

Omni-modal large language models (OmniLLMs) support unified audio-visual understanding. However, processing large numbers of audio and video tokens causes substantial computational and memory costs, making efficient token compression essential. Audio and video can provide unique or repeated information at different moments, while existing methods typically apply one fixed strategy throughout a sequence. Guiding one modality with the other may discard information unique to each modality when they are complementary, while compressing them separately may preserve redundant information when they overlap. Such fixed strategies fail to adapt to this changing correlation. To address this mismatch, we propose **OmniTrim**, a training-free framework for adaptive audio-visual token compression. Specifically, Information-Aware Budgeting allocates the token budget across temporal chunks according to their audio-visual activity and temporal variation. Correlation-Aware Selection then adaptively chooses between separate selection and joint token merging based on the local audio-visual correlation, preserving complementary information and removing redundancy. Experiments on six audio-visual benchmarks demonstrate that OmniTrim outperforms existing OmniLLM compression baselines. At 25% token retention, it preserves 98.8% of the full-token performance on Qwen2.5-Omni-7B, while delivering a 4.35× prefill speedup, increasing decoding throughput by 55.3%, and maintaining the lowest peak GPU memory usage and end-to-end latency.

1 INTRODUCTION

Omnimodal large language models (OmniLLMs) jointly process text, audio, and video, integrating visual content with speech and environmental sounds for audio-visual question answering and real-time interaction (Xie & Wu, 2024; Xu et al., 2025a; Tang et al., 2025; Xu et al., 2025b). However, dense video frames and long audio sequences produce large numbers of tokens, leading to substantial computation and memory costs during LLM prefill. Token compression offers a practical way to reduce these costs and has achieved substantial progress in the visual domain (Chen et al., 2024;

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Varying audio–visual correlation

(b) Complementary and redundant chunks

(c) Fixed selection

(d) Adaptive selection

Figure 2: (a) A schematic timeline illustrates variation in local audio–visual correlation. (b) This variation produces complementary and redundant chunks with different information patterns. (c) Fixed selection can discard modality-specific information or preserve duplicated information. (d) OmniTrim adapts the selection mode to the local correlation of each chunk.

Shang et al., 2025; Zhang et al., 2025b; Yang et al., 2025; Tao et al., 2025; Fu et al., 2025b; Alvar et al., 2025; Du et al., 2026; Luo et al., 2026).

In OmniLLMs, audio and video do not contribute the same type of information throughout a sequence. As their content evolves, the correlation between modalities is dynamic and varies across chunks. In some chunks, the two modalities are complementary, with each providing information absent from the other. In other chunks, they are redundant, with both describing the same content. Although achieving promising results, existing methods generally apply a fixed compression strategy throughout a sequence and do not adapt it to these changes. Methods such as OmniZip, OmniSIFT, and OmniFit use one modality to guide the other (Tao et al., 2026a; Ding et al., 2026; Wang et al., 2026), assuming that it captures the information needed from the other modality. In complementary chunks, this strategy may discard information unique to one modality. By contrast, methods such as OmniScope and OmniSelect compress the two modalities separately (Su et al., 2026; Yang et al., 2026). In redundant chunks, this strategy may retain redundant information. Therefore, a fixed strategy cannot adapt to changing audio-visual correlation, as illustrated in Figure 2.

To address this problem, we introduce **OmniTrim**, a training-free two-stage framework for adaptive audio-visual token compression. First, **Information-Aware Budgeting** distributes the global token budget across temporal chunks according to audio-visual activity and content variation, assigning more tokens to information-rich chunks. Within each chunk, **Correlation-Aware Selection** adapts compression to the local audio-visual correlation. It selects tokens separately in complementary chunks, where the modalities provide distinct information, and jointly in redundant chunks, where they convey overlapping information, preserving modality-specific information while reducing cross-modal redundancy, as illustrated in Figure 2(d). Together, the modules determine *where* to allocate tokens and *how* to compress each chunk. Across six audio-visual benchmarks, OmniTrim consistently outperforms existing OmniLLM compression methods in average performance as illustrated in Figure 1. At 25% token retention on Qwen2.5-Omni-7B, it preserves 98.8% of the uncompressed performance, cuts prefill time by 77.0%, and raises decoding throughput by 55.3%, while maintaining the lowest peak GPU memory usage and end-to-end latency.

Our contributions are summarized as follows:

- We reveal how audio-visual correlation changes over time, where audio and video provide distinct information in complementary chunks and repeated information in redundant chunks throughout the same sequence.

- We introduce OmniTrim, a training-free framework that allocates tokens according to chunk-level information content and adapts between separate and joint compression according to local audio-visual correlation.
- Experiments show that OmniTrim achieves state-of-the-art performance for audio-visual token compression across benchmarks and model scales, while preserving nearly all of the full-token performance and improving inference efficiency.

2 Related Work

2.1 Omnimodal Large Language Models

Omnimodal large language models (OmniLLMs) extend vision-language models by jointly processing text, visual, and audio inputs. They support applications such as audio-visual question answering, real-time spoken interaction, and multimodal generation. Representative open systems such as Mini-Omni2 (Xie & Wu, 2024), VITA-1.5 (Fu et al., 2025a), and Baichuan-Omni-1.5 (Li et al., 2025) add end-to-end speech interaction to vision-language models. More recent models, including Qwen2.5-Omni (Xu et al., 2025a), video-SALMONN 2 (Tang et al., 2025), Qwen3-Omni (Xu et al., 2025b), and Dynin-Omni (Kim et al., 2026), improve synchronized audio-video understanding and move toward unified perception and generation across modalities. Despite this progress, dense video frames and long audio sequences produce long multimodal token sequences, causing computation and memory costs to grow with input duration. Token compression directly shortens the audio-visual context and offers a practical way to speed up OmniLLM inference.

2.2 Token Compression in Multimodal Models

In the visual domain, token compression is widely studied (Shao et al., 2026). For images, methods reduce redundancy by merging similar tokens, using text to guide token selection, and pruning tokens based on attention (Bolya et al., 2023; Shang et al., 2025; Chen et al., 2024; Zhang et al., 2025b; Yang et al., 2025; Tan et al., 2025). For videos, methods also exploit spatial and temporal redundancy across frames (Fan et al., 2026; Tang et al., 2026; Cho et al., 2026), including unified spatiotemporal allocation under an ultra-low retention budget (Huang et al., 2025; Zhuang et al., 2025; Tao et al., 2025; Shao et al., 2025; Fu et al., 2025b; Du et al., 2026; Shen et al., 2025b). In the audio domain, methods reduce the number of tokens produced by high-rate audio encoders through adjacent token merging and importance-based pruning (Li et al., 2023; Lee & Lee, 2025; Lin et al., 2025). These modality-specific methods show that each modality contains substantial redundancy, but they do not address how audio and video should be compressed together under a shared budget. Recent methods therefore study joint compression of synchronized audio and video (Li & Xiong, 2026). OmniZip (Tao et al., 2026a) uses audio tokens to guide video pruning, whereas OmniSIFT (Ding et al., 2026) prunes video tokens based on spatial and temporal redundancy before using visual information to select audio tokens. OmniFit (Wang et al., 2026) changes compression budgets across decoder layers and between modalities. OmniSelect (Yang et al., 2026) chooses a compression strategy and allocates temporal-group budgets based on the relevance of each modality to the query. OmniScope (Su et al., 2026) uses the query to estimate audio and video relevance separately and assigns modality-specific budgets. Omni-Prune (Zhong et al., 2026) scores audio and video tokens on a common scale and keeps related token pairs within adaptive temporal windows. However, existing methods do not account for how the audio-visual correlation changes across temporal chunks. This can cause modality-specific information to be discarded when the modalities provide different information and redundant information to be retained when they convey similar content. To address this limitation, OmniTrim adapts token budgets and compression strategies to each chunk's information content and local audio-visual correlation.

3 Method

3.1 Overview

As illustrated in Figure 3, OmniTrim consists of two stages. **Information-Aware Budgeting** allocates the shared budget across temporal chunks according to their activity and temporal variation.

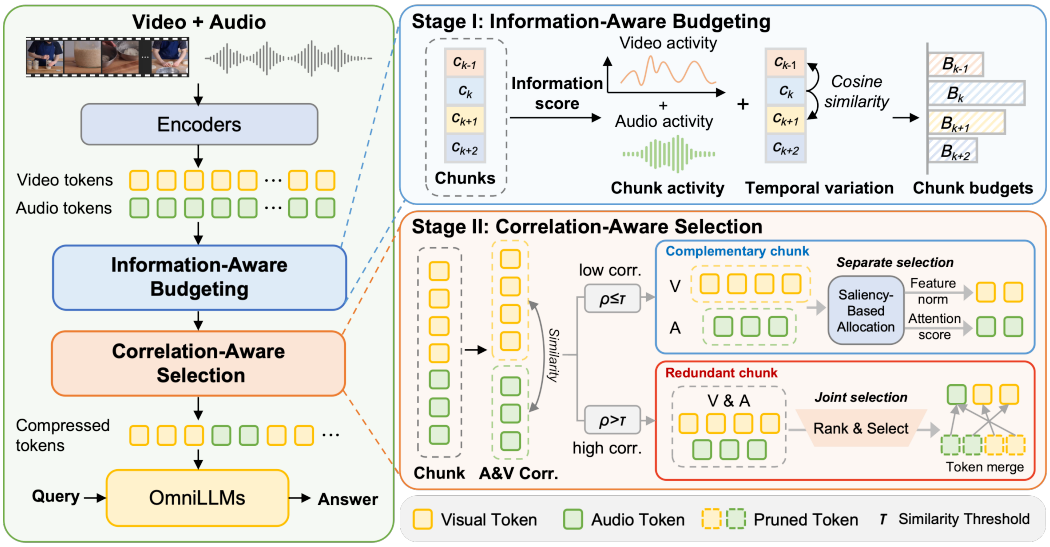


Figure 3: **Overall framework of OmniTrim.** Information-Aware Budgeting allocates the shared token budget across temporal chunks based on audio-visual activity and temporal variation. Correlation-Aware Selection uses the local audio-visual correlation to select tokens separately in complementary chunks and jointly in redundant chunks, followed by token merging for joint selection. The compressed audio-visual tokens are combined with text tokens for OmniLLM inference.

Correlation-Aware Selection then adapts the compression mode to the local audio-visual similarity, selecting the two modalities separately when they provide different information and jointly when they contain overlapping information. Together, the two stages adapt token compression to both temporal information content and local audio-visual correlation. The retained audio and visual tokens are then combined with the text tokens and passed to the OmniLLM for subsequent inference.

3.2 Information-Aware Budgeting

Long audio-visual sequences contain static and repeated content, with only occasional brief audio events and visual changes. A uniform retention ratio may assign too many tokens to redundant intervals and too few to informative moments. Information-Aware Budgeting addresses this imbalance by using token activity and temporal variation to redistribute the shared budget over time. The visual and audio modalities contain N_v and N_a tokens, respectively. Given a retention ratio $r \in (0, 1]$, the shared audio-visual budget is $B = \mathrm{round}(r(N_v + N_a))$, while text tokens remain unchanged. The audio-visual sequence is partitioned into K temporally aligned chunks. The following definitions are given for a generic chunk, with the chunk index omitted for notational simplicity. Within a chunk, modality $m \in \{v, a\}$ contains n_m tokens denoted by $z_{m,i}$, and its mean-pooled representation is $\bar{z}_m$. The chunk activity A_m and temporal variation V_m are defined as follows:

$$A_m = \frac{1}{n_m} \sum_{i=1}^{n_m} \|z_{m,i}\|_2, \qquad V_m = 1 - \frac{1}{|\mathcal{N}|} \sum_{j \in \mathcal{N}} \cos(\bar{z}_m, \bar{z}_m^j), \tag{1}$$

where $\mathcal{N}$ contains the temporal neighbors of the current chunk, and $\bar{z}_m^j$ denotes the mean-pooled representation of modality m in neighboring chunk j. A_m measures the average response magnitude of the current chunk (Oyama et al., 2023). V_m captures the temporal distinctiveness of the current chunk by measuring how closely its representation aligns with those of nearby chunks. Lower similarity indicates that the chunk contains information that differs from its local temporal context. We combine the normalized activity $\widetilde{A}_m$ and temporal variation $\widetilde{V}_m$ into a modality-specific information score I_m and fuse the visual and audio scores I_v and I_a into a chunk-level information score I that captures the information content of both modalities:

$$I_m = \alpha \widetilde{A}_m + (1 - \alpha)\widetilde{V}_m, \qquad I = \mathrm{Norm}(\lambda I_v + (1 - \lambda) I_a)\,. \tag{2}$$

Here, $\alpha \in [0,1]$ balances activity and temporal variation, $\lambda \in [0,1]$ balances visual and audio information, and both the tilde notation and Norm denote min–max normalization across chunks. We adjust the global retention ratio r according to each chunk's score I relative to the sequence mean $\overline{I}$, assigning higher retention ratios to chunks with above-average scores:

$$\widehat{r} = r\left[1 + \gamma\left(I - \overline{I}\right)\right], \tag{3}$$

where $\gamma \geq 0$ controls the adjustment strength. Furthermore, we smooth the retention ratios in both temporal directions and average the results, then limit differences between neighboring chunks to avoid abrupt changes in token allocation. The resulting chunk budgets B_k are adjusted to satisfy the shared constraint $\sum_{k=1}^{K} B_k = B$, where k indexes the chunks. This allocation preserves more tokens around informative moments while compressing static and repetitive intervals aggressively.

3.3 Correlation-Aware Selection

Temporal information density determines how the shared token budget is distributed across chunks. However, the audio-visual correlation varies across chunks. In some chunks, the two modalities provide complementary information, whereas in others they describe overlapping content. After assigning a budget to each chunk, we determine how that budget should be used according to the local audio-visual correlation. Separate selection preserves modality-specific information when the modalities differ, whereas joint token merging reduces duplicated information when they overlap. We estimate the local correlation from the mean-pooled audio and visual representations using an audio-visual similarity score ρ

$$\rho = \frac{1}{2}\left(1 + \cos(\overline{z}_v, \overline{z}_a)\right), \tag{4}$$

where $\rho \in [0,1]$ denotes the audio-visual similarity score of the current chunk and serves as the routing signal. A similarity threshold $\tau \in [0,1]$ separates the two selection modes. To characterize the information contributed by each modality within the chunk, we compute the saliency S_m for each modality $m \in \{v, a\}$:

$$S_m = \widetilde{A}_m + \eta \widetilde{V}_m, \qquad m \in \{v, a\}, \tag{5}$$

where η controls the contribution of temporal variation to the saliency score. As in the information score defined above, activity and temporal variation jointly indicate informative content, here guiding token allocation between modalities within each chunk. At the token level, we further measure visual importance by feature magnitude and audio importance by encoder attention:

$$s_{v,i} = \|z_{v,i}\|_2, \qquad s_{a,i} = h_{a,i}, \tag{6}$$

where $h_{a,i}$ denotes the attention-derived importance of audio token i from the audio encoder.

Separate selection for low-similarity chunks. When $\rho \leq \tau$, the modalities are more likely to provide distinct information. The chunk budget is allocated to the visual and audio modalities in proportion to their saliency scores S_v and S_a defined in Eq. 5. Visual and audio tokens are then selected independently within their respective budgets. This separate selection allows each modality to preserve its own information when the audio and visual modalities differ.

Joint selection for high-similarity chunks. Conversely, when $\rho > \tau$, the two modalities are more likely to contain overlapping information. The joint selection approach is employed to address this scenario. We rank the combined audio-visual candidates by their token-level importance and jointly select them under the chunk budget. To preserve information from discarded tokens, we assign each to its most similar selected token and merge their features using similarity-based weights, with higher weights for more similar tokens, without increasing the sequence length. By switching between modality-specific selection and joint token merging according to the local audio-visual correlation, the design preserves modality-specific information in complementary chunks while removing cross-modal redundancy in redundant chunks.

4 Experiments

In this section, we conduct extensive experiments across multiple audio-visual understanding benchmarks and two model scales. We first introduce the experimental settings, then present the main results, and finally discuss the ablation studies and efficiency analysis.

4.1 Experimental Settings

Benchmarks. We evaluate OmniTrim on six audio-visual understanding benchmarks. They include ShortVid-Bench (Ge et al., 2025), WorldSense (Hong et al., 2026), OmniVideoBench (Li et al., 2026), Video-Holmes (Cheng et al., 2025), JointAVBench (Chao et al., 2026), and LVOmniBench (Tao et al., 2026b). Together, these benchmarks cover short and long videos and test temporal reasoning, complex inference, and audio-visual integration.

Compared methods. We compare OmniTrim with four training-free token compression methods. FastVID (Shen et al., 2025a) partitions videos into temporal segments and performs density-based token pruning to preserve spatial and temporal information. DyCoke (Tao et al., 2025) combines inter-frame token merging with dynamic KV cache pruning. OmniZip (Tao et al., 2026a) uses audio importance to guide video token pruning, whereas OmniScope (Su et al., 2026) compresses audio and video independently. The uncompressed baseline provides the full-token reference.

Implementation Details. We evaluate OmniTrim with Qwen2.5-Omni-7B and Qwen2.5-Omni-3B (Xu et al., 2025a). All experiments are run on a single NVIDIA A800 GPU. We cap each video at 768 sampled frames and use the same prompts, generation settings, answer parsing, and scoring protocol for all methods (Zhang et al., 2025a). For the video compression baselines, we apply FastVID and DyCoke to both the audio and video modalities, denoted as V&A, while retaining their original settings. For the omnimodal baselines, we reproduce OmniZip and OmniScope using their official implementations and paper-reported hyperparameters. When a target retention ratio is not reported, we interpolate or extrapolate their modality-specific pruning ratios following the original settings. For Information-Aware Budgeting, we set the token activity weight to $\alpha = 0.65$, the visual information weight to $\lambda = 0.55$, and the budget adjustment strength to $\gamma = 0.8$. For Correlation-Aware Selection, we set the correlation threshold to $\tau = 0.66$ and the temporal variation weight to $\eta = 0.25$. In all experiments, we use FlashAttention (Dao, 2024) to reduce memory usage.

4.2 Main Results

State-of-the-Art Performance. Table 1 compares OmniTrim with representative token compression methods under constrained retention budgets. On Qwen2.5-Omni-7B, OmniTrim achieves the best average performance at every retention ratio. At 45% retention, it preserves 100.1% of the uncompressed performance while removing more than half of the tokens, and even surpasses the uncompressed baseline on JointAVBench and LVOmniBench. As token retention decreases to 35% and 25%, OmniTrim retains 99.2% and 98.8% of the uncompressed performance, respectively. The comparison with the baselines further explains this advantage. OmniZip uses audio importance to guide video pruning, whereas OmniScope compresses audio and video independently. OmniTrim instead allocates tokens according to local information content and switches between separate and joint selection based on audio-visual correlation. At 25% retention, it exceeds OmniZip and OmniScope by 2.2 and 2.3 percentage points in relative performance, respectively. The largest accuracy gains over the strongest baseline occur on Video-Holmes and OmniVideoBench, reaching 1.8 and 1.3 points, respectively. FastVID and DyCoke address redundancy within each modality but exhibit larger performance losses when applied to both audio and video under tighter budgets. From 45% to 25% retention, their relative performance drops by 3.2 and 5.9 percentage points, respectively, compared with only 1.3 for OmniTrim. These results demonstrate that OmniTrim consistently achieves higher average accuracy than existing compression methods and better preserves performance across diverse audio-visual benchmarks, even when only a small fraction of the original tokens is retained. Furthermore, experiments on Qwen2.5-Omni-3B show that OmniTrim remains effective across model sizes. It achieves the best average performance at all three retention ratios, preserving 99.8% of the uncompressed performance at 45% retention and 97.7% at 25% retention. OmniTrim thus maintains strong performance under aggressive token compression across both model sizes and the evaluated audio-visual benchmarks.

4.3 Ablation Studies

Information-Aware Budgeting. Table 2 shows that the performance gap between full OmniTrim and w/o IAB increases as the budget tightens, from 0.8 relative-performance percentage points at

Table 1: **Comparison with state-of-the-art methods.** We evaluate the full-token setting and compressed inputs at $r \in \{45\%, 35\%, 25\%\}$. Avg. is the mean score across the six benchmarks, and Rel. reports performance relative to the corresponding Vanilla setting. The **best** results in each setting are bolded, and the second-best are underlined.

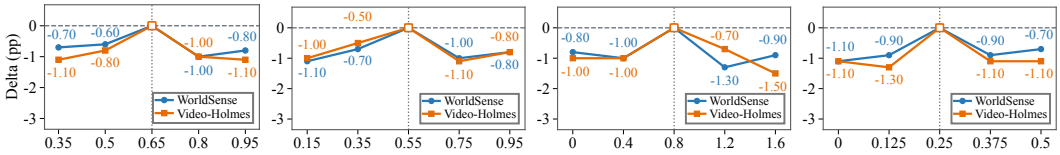
Method	Retention Ratio r	Short Vid	World Sense	OmniVideo Bench	Video-Holmes	JointAV Bench	LVOmni Bench	Avg.	Rel.
Qwen2.5-Omni-7B									
Vanilla	100%	70.9	46.9	36.3	46.9	62.8	31.7	49.2	100.0%
FastVID (V&A)	45%	69.7	42.9	33.6	45.1	62.7	31.3	47.6	96.2%
DyCoke (V&A)	45%	69.4	44.9	34.5	45.3	63.0	**32.8**	48.3	98.3%
OmniZip	45%	69.2	46.2	**36.2**	45.9	62.8	31.8	48.7	99.0%
OmniScope	45%	69.2	**46.6**	35.7	44.2	62.2	31.4	48.2	98.0%
OmniTrim	45%	**70.3**	46.5	36.0	**46.3**	**63.5**	**32.8**	**49.2**	**100.1%**
FastVID (V&A)	35%	68.3	41.7	33.0	44.1	61.8	32.1	46.8	95.0%
DyCoke (V&A)	35%	**69.5**	43.3	34.1	44.5	62.7	32.2	47.7	96.9%
OmniZip	35%	67.9	45.4	**36.2**	45.6	62.7	**32.3**	48.4	98.6%
OmniScope	35%	68.8	**45.7**	36.1	43.8	61.5	31.0	47.8	97.2%
OmniTrim	35%	69.1	**45.7**	36.1	**46.4**	**63.1**	32.1	**48.8**	**99.2%**
FastVID (V&A)	25%	67.1	40.9	31.7	42.7	61.7	31.6	45.9	93.0%
DyCoke (V&A)	25%	67.0	40.1	32.0	42.0	61.2	31.5	45.6	92.4%
OmniZip	25%	67.4	44.7	34.8	43.9	62.3	31.9	47.5	96.6%
OmniScope	25%	**68.0**	44.7	35.1	43.5	61.7	31.8	47.5	96.5%
OmniTrim	25%	**68.0**	**45.0**	**36.4**	**45.7**	**63.0**	**32.6**	**48.5**	**98.8%**
Qwen2.5-Omni-3B									
Vanilla	100%	70.0	46.7	34.2	43.1	52.7	33.3	46.7	100.0%
FastVID (V&A)	45%	68.9	42.8	33.1	41.5	50.3	34.1	45.1	96.8%
DyCoke (V&A)	45%	68.9	43.9	33.7	**42.4**	50.6	32.6	45.4	97.3%
OmniZip	45%	68.7	46.3	33.8	41.8	51.7	33.6	46.0	98.7%
OmniScope	45%	68.5	**46.5**	32.8	41.8	52.1	32.4	45.7	97.8%
OmniTrim	45%	**69.1**	45.8	**33.9**	42.1	**52.2**	**35.3**	**46.4**	**99.8%**
FastVID (V&A)	35%	**68.9**	42.2	31.8	40.4	49.1	33.0	44.2	94.7%
DyCoke (V&A)	35%	68.7	42.5	32.4	40.6	49.7	33.1	44.5	95.3%
OmniZip	35%	68.7	**46.2**	33.4	**41.5**	51.4	33.4	45.8	98.2%
OmniScope	35%	67.8	45.9	32.7	**41.5**	51.6	32.7	45.4	97.2%
OmniTrim	35%	**68.9**	44.9	**34.2**	**41.5**	**51.8**	**34.2**	**45.9**	**98.7%**
FastVID (V&A)	25%	67.0	40.5	31.0	39.4	48.0	32.3	43.0	92.1%
DyCoke (V&A)	25%	66.5	39.9	32.0	39.4	48.0	32.5	43.1	92.4%
OmniZip	25%	**68.9**	44.7	33.1	**41.4**	50.5	33.4	45.3	97.2%
OmniScope	25%	67.7	**45.1**	32.0	40.3	**51.5**	32.8	44.9	96.1%
OmniTrim	25%	67.5	44.3	**33.4**	41.3	51.4	**34.6**	**45.4**	**97.7%**

45% retention to 1.4 at 35% and 2.0 at 25%. At 25% retention, the full model scores higher on all six benchmarks, with the largest accuracy gap of 1.9 points on ShortVid. We further examine temporal budget smoothing within IAB. The w/o smoothing variant remains close to the full model at 45% retention, but falls behind by 0.7 relative-performance percentage points at 35% and 1.7 at 25%. These trends favor information-aware allocation with temporal smoothing when only a small fraction of the original audio-visual tokens can be retained.

Correlation-Aware Selection. Adaptive selection outperforms Separate-only and Joint-only on average at all three retention ratios. At 25% retention, these fixed policies retain 96.6% and 94.4% of the uncompressed performance, respectively, compared with 98.8% for full OmniTrim. The growing gap over Separate-only suggests that reducing cross-modal redundancy becomes more important under tighter budgets. The full model also outperforms both policies on every benchmark at this ratio. Although Joint-only leads on LVOmniBench at higher retention, its lower average performance favors adapting selection to local audio-visual correlation.

Table 2: **Module ablation on Qwen2.5-Omni-7B.** IAB denotes Information-Aware Budgeting. The w/o smoothing variant disables temporal smoothing of chunk retention ratios. Separate-only selects audio and visual tokens independently, whereas Joint-only selects them jointly within each chunk under the shared audio-visual token budget.

Retain	Variant	Short Vid	World Sense	OmniVideo Bench	Video-Holmes	JointAV Bench	LVOmni Bench	Avg.	Rel.
45%	**Full OmniTrim**	**70.3**	**46.5**	36.0	**46.3**	**63.5**	32.8	**49.2**	**100.1%**
	w/o IAB	69.8	45.9	36.0	46.2	63.3	32.1	48.9	99.3%
	w/o smoothing	69.9	46.1	36.5	**46.3**	63.2	32.6	49.1	99.9%
	Separate-only	69.3	46.3	**36.7**	46.2	63.2	31.5	48.9	99.3%
	Joint-only	67.3	42.5	31.7	42.9	61.0	**33.8**	46.5	94.7%
35%	**Full OmniTrim**	**69.1**	**45.7**	**36.1**	**46.4**	**63.1**	32.1	**48.8**	**99.2%**
	w/o IAB	68.4	45.6	35.6	45.7	62.9	30.9	48.2	97.8%
	w/o smoothing	68.4	45.5	35.5	46.0	**63.1**	32.0	48.4	98.5%
	Separate-only	68.1	45.6	35.4	45.8	63.0	32.1	48.3	98.4%
	Joint-only	66.3	42.1	30.8	43.0	60.9	**32.3**	45.9	93.1%
25%	**Full OmniTrim**	**68.0**	**45.0**	**36.4**	**45.7**	**63.0**	**32.6**	**48.5**	**98.8%**
	w/o IAB	66.1	44.3	35.8	44.8	62.1	31.8	47.5	96.8%
	w/o smoothing	66.0	44.6	35.3	45.0	62.1	32.5	47.6	97.1%
	Separate-only	67.3	44.4	35.3	44.8	61.9	31.4	47.5	96.6%
	Joint-only	63.2	44.0	33.7	44.2	61.0	31.4	46.3	94.4%



(a) Ablation on α **(b)** Ablation on λ **(c)** Ablation on γ **(d)** Ablation on η

Figure 4: **Parameter sensitivity at 35% token retention.** Results are reported as score changes relative to the default setting.

Correlation Threshold. The threshold τ determines which chunks undergo separate or joint selection according to their audio-visual correlation. Figure 5 shows that increasing τ shifts more chunks to separate selection on ShortVid at 45% retention, but accuracy does not increase monotonically with this shift. It peaks at 70.3% when $\tau = 0.66$, where 86.9% of chunks use separate selection and the remaining chunks use joint selection. Routing all chunks separately reduces accuracy by 1.0 point, while extending joint selection to more chunks at lower thresholds also lowers accuracy. At higher thresholds, accuracy stabilizes as nearly all chunks follow the separate route and the routing distribution changes little. Thus, joint selection is beneficial when applied selectively to chunks with higher audio-visual correlation. The results favor preserving modality-specific information in complementary chunks while jointly compressing redundant chunks.

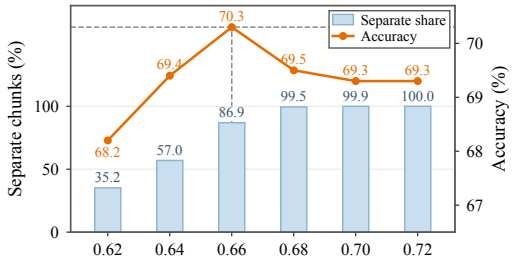


Figure 5: **Ablation study on τ.** Separate-chunk share and accuracy on ShortVid at 45% retention as τ varies.

Information Score Weights. Figure 4 examines the token activity weight α and the visual information weight λ. The weight α balances the current activity of a chunk against its temporal change, while λ balances the visual and audio contributions to the fused score. At 35% retention, the reported scores are highest at the default settings on both benchmarks, with other tested settings

Table 3: **Inference efficiency comparison on Video-Holmes.** Under the same token budget, OmniTrim achieves higher accuracy than the baselines with faster prefill and lower peak GPU memory. With fewer tokens, it retains this accuracy advantage while further improving decoding throughput and reducing end-to-end latency.

Method	Retention Ratio r	GPU Mem. (GB) ↓	Prefill (ms) ↓	Throughput (token/s) ↑	E2E Latency (s) ↓	Acc. ↑
Vanilla	100%	26.96	1115.0 (1.00×)	24.40 (1.00×)	3.303	46.9
FastVID (V&A)	35%	23.98	377.1 (2.96×)	10.07 (0.41×)	3.400	44.1
OmniZip	35%	23.89	432.1 (2.58×)	35.09 (1.44×)	2.758	45.6
OmniScope	35%	24.79	453.3 (2.46×)	33.73 (1.38×)	3.972	43.8
OmniTrim	35%	23.25	361.0 (3.09×)	34.79 (1.43×)	2.833	**46.4**
OmniTrim	25%	**22.65**	**256.4 (4.35×)**	**37.90 (1.55×)**	**2.737**	45.7

showing reductions of 0.5–1.1 percentage points. This pattern is consistent with the complementary roles of activity and temporal variation, which capture response strength within each chunk and content changes across neighboring chunks, respectively.

Budget Redistribution Strength. The adjustment strength γ controls how much the token budget varies across chunks. At 35% retention, the reported scores are highest at $\gamma = 0.8$ on both benchmarks. When γ is too small, informative chunks receive little additional budget. When it is too large, concentrating tokens in these chunks can leave too few for surrounding context. A moderate value can therefore give informative chunks more tokens while preserving the surrounding context needed to interpret events across the sequence.

Modality Saliency Weight. The weight η controls the contribution of temporal variation to modality saliency, guiding the division of each chunk's budget between audio and video. At 35% retention, the reported accuracy is highest at $\eta = 0.25$ on both benchmarks, with lower scores at all other tested values. Setting $\eta = 0$ ignores temporal changes in this allocation, while an excessive weight can overemphasize change relative to response strength. A moderate contribution can therefore help preserve changing content without discounting informative content that remains stable, allowing both signals to guide the allocation within each chunk.

4.4 Efficiency Analysis

Table 3 reports inference efficiency and accuracy on Video-Holmes under identical input and hardware settings. Under the same token budget, OmniTrim achieves higher accuracy than the baselines while reducing prefill time and peak GPU memory. With fewer retained tokens, OmniTrim still exceeds the accuracy of the baselines while further improving decoding throughput and reducing end-to-end latency. Relative to the uncompressed baseline, this setting provides a 4.35× prefill speedup and a 16.0% memory reduction. These results demonstrate the effectiveness of OmniTrim in preserving task performance while improving inference efficiency.

5 Conclusion

In this paper, we propose OmniTrim, a training-free framework for adaptive audio-visual token compression in OmniLLMs. Audio and video provide complementary or redundant information at different moments, motivating compression strategies that adapt to these changing relationships. By allocating tokens according to local information content and adapting selection to audio-visual correlation, OmniTrim preserves modality-specific information in complementary chunks and reduces cross-modal redundancy in redundant chunks. Experiments demonstrate state-of-the-art performance across audio-visual benchmarks, establishing OmniTrim as an effective approach to improving OmniLLM inference efficiency while preserving task performance.

AI USE STATEMENT

We used generative AI tools to assist with manuscript writing and language polishing, including improving grammar, clarity, and readability, to support literature retrieval and the discovery of related work, and to assist with coding. The authors reviewed and revised the AI-assisted text, checked the cited sources and their relevance against the original publications, and manually reviewed all AI-assisted code. The authors take full responsibility for the final manuscript and code, including the accuracy of the claims, results, and references.

REFERENCES

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. DivPrune: Diversity-based visual token pruning for large multimodal models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9392–9401. IEEE, 2025.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023.

Jianghan Chao, Jianzhang Gao, Wenhui Tan, Yuchong Sun, Ruihua Song, and Liyun Ru. JointAVBench: A benchmark for joint audio-visual reasoning evaluation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 84035–84062, 2026.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. 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 (ECCV)*, pp. 19–35, 2024. doi: 10.1007/978-3-031-73004-7_2.

Junhao Cheng, Yuying Ge, Teng Wang, Yixiao Ge, Jing Liao, and Ying Shan. Video-Holmes: Can MLLM think like Holmes for complex video reasoning? *arXiv preprint arXiv:2505.21374*, 2025.

Janghoon Cho, Jungsoo Lee, Munawar Hayat, Kyuwoong Hwang, Fatih Porikli, and Sungha Choi. FLoC: Facility location-based efficient visual token compression for long video understanding. In *International Conference on Learning Representations*, volume 2026, pp. 102973–102996, 2026.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, volume 2024, pp. 35549–35562, 2024.

Yue Ding, Yiyan Ji, Jungang Li, Xuyang Liu, Xinlong Chen, Junfei Wu, Bozhou Li, Bohan Zeng, Yang Shi, Yushuo Guan, Yuanxing Zhang, Jiaheng Liu, Qiang Liu, Pengfei Wan, and Liang Wang. OmniSIFT: Modality-asymmetric token compression for efficient omni-modal large language models. In *Forty-third International Conference on Machine Learning*, 2026.

Junhao Du, Jialong Xue, Anqi Li, Jincheng Dai, and Guo Lu. Unified spatiotemporal token compression for video-LLMs at ultra-low retention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17661–17671, June 2026.

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 *International Conference on Learning Representations*, volume 2026, pp. 98730–98759, 2026.

Chaoyou Fu, Haojia Lin, Xiong Wang, Yifan Zhang, Yunhang Shen, Xiaoyu Liu, Haoyu Cao, Zuwei Long, Heting Gao, Ke Li, Long Ma, Xiawu Zheng, Rongrong Ji, Xing Sun, Caifeng Shan, and Ran He. VITA-1.5: Towards GPT-4o level real-time vision and speech interaction. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 75300–75320. Curran Associates, Inc., 2025a. doi: 10.52202/085713-2529.

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22654–22663. IEEE, 2025b.

Yuying Ge, Yixiao Ge, Chen Li, Teng Wang, Junfu Pu, Yizhuo Li, Lu Qiu, Jin Ma, Lisheng Duan, Xinyu Zuo, et al. ARC-Hunyuan-Video-7B: Structured video comprehension of real-world shorts. *arXiv preprint arXiv:2507.20939*, 2025.

Jack Hong, Shilin Yan, Jiayin Cai, Xiaolong Jiang, Yao Hu, and Weidi Xie. WorldSense: Evaluating real-world omnimodal understanding for multimodal LLMs. In *International Conference on Learning Representations*, volume 2026, pp. 52423–52443, 2026.

Xiaohu Huang, Hao Zhou, and Kai Han. PruneVid: Visual token pruning for efficient video large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19959–19973, 2025.

Jaeik Kim, Woojin Kim, Jihwan Hong, Yejoon Lee, Sieun Hyeon, Mintaek Lim, Yunseok Han, Dogeun Kim, Hoeun Lee, Hyunggeun Kim, and Jaeyoung Do. Dynin-Omni: Omnimodal unified large diffusion language model. *arXiv preprint arXiv:2604.00007*, 2026.

Taehan Lee and Hyukjun Lee. Token pruning in audio transformers: Optimizing performance and decoding patch importance. In *ECAI 2025*. IOS Press, October 2025. doi: 10.3233/faia251136.

Caorui Li, Yu Chen, Yiyan Ji, Jin Xu, Zhenyu Cui, Shihao Li, Yuanxing Zhang, Zhenghao Song, Dingling Zhang, Ying He, et al. OmniVideoBench: Towards audio-visual understanding evaluation for omni MLLMs. In *International Conference on Learning Representations*, volume 2026, pp. 138214–138236, 2026.

Song Li and Yongping Xiong. PRIM: Cooperative dynamic token compression for efficient large multimodal models. In *Forty-third International Conference on Machine Learning*, 2026.

Yadong Li, Jun Liu, Tao Zhang, Song Chen, Tianpeng Li, Zehuan Li, Lijun Liu, Lingfeng Ming, Guosheng Dong, Da Pan, et al. Baichuan-Omni-1.5 technical report. *arXiv preprint arXiv:2501.15368*, 2025.

Yuang Li, Yu Wu, Jinyu Li, and Shujie Liu. Accelerating Transducers through Adjacent Token Merging. In *Interspeech 2023*, pp. 1379–1383, 2023. doi: 10.21437/Interspeech.2023-599.

Yueqian Lin, Yuzhe Fu, Jingyang Zhang, Yudong Liu, Jianyi Zhang, Jingwei Sun, Hai Helen Li, and Yiran Chen. SpeechPrune: Context-aware token pruning for speech information retrieval. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2025.

Yongdong Luo, Wang Chen, Weizhong Huang, Shukang Yin, Haojia Lin, Jinfa Huang, Chaoyou Fu, Jiayi Ji, Xiawu Zheng, and Jiebo Luo. QuoTA: Query-oriented token assignment via CoT query decouple for long video comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24160–24168, 2026.

Momose Oyama, Sho Yokoi, and Hidetoshi Shimodaira. Norm of word embedding encodes information gain. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2108–2130, 2023.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive token reduction for efficient large multimodal models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22857–22867. IEEE, 2025.

Kele Shao, Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. HoliTom: Holistic token merging for fast video large language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 135547–135570, 2025.

Kele Shao, Keda Tao, Kejia Zhang, Sicheng Feng, Mu Cai, Yuzhang Shang, Haoxuan You, Can Qin, Yang Sui, and Huan Wang. A survey of token compression for efficient multimodal large language models. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. Survey Certification.

Leqi Shen, Guoqiang Gong, Tao He, Yifeng Zhang, Pengzhang Liu, Sicheng Zhao, et al. FastVID: Dynamic density pruning for fast video large language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 123553–123581, 2025a.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, Zhuang Liu, Hu Xu, Hyunwoo J. Kim, Bilge Soran, Raghuraman Krishnamoorthi, Mohamed Elhoseiny, and Vikas Chandra. LongVU: Spatiotemporal adaptive compression for long video-language understanding. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 54582–54599, 2025b.

Jinsen Su, Yongdong Luo, Yuexiao Ma, Yibo Hu, Meiguang Jin, and Xiawu Zheng. OmniScope: Modality-decoupled token compression for omnimodal large language models. *arXiv preprint arXiv:2607.23193*, 2026.

Xudong Tan, Peng Ye, Chongjun Tu, Jianjian Cao, Yaoxin Yang, Lin Zhang, Dongzhan Zhou, and Tao Chen. TokenCarve: Information-preserving visual token compression in multimodal large language models. *arXiv preprint arXiv:2503.10501*, 2025.

Changli Tang, Yixuan Li, Yudong Yang, Jimin Zhuang, Guangzhi Sun, Wei Li, Zejun Ma, and Chao Zhang. video-SALMONN 2: Caption-enhanced audio-visual large language models. *arXiv preprint arXiv:2506.15220*, 2025.

Chong Tang, Sannara Ek, Dirk Koch, Robert Mullins, Alex Weddell, and Jagmohan Chauhan. SURGE: Surprise-guided token reduction for efficient video understanding with VLMs. In *International Conference on Learning Representations*, volume 2026, pp. 3849–3868, 2026.

Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. DyCoke: Dynamic compression of tokens for fast video large language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18992–19001. IEEE, 2025.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17682–17692, 2026a.

Keda Tao, Yuhua Zheng, Jia Xu, Wenjie Du, Kele Shao, Hesong Wang, Xueyi Chen, Xin Jin, Junhan Zhu, Bohan Yu, et al. LVOmniBench: Pioneering long audio-video understanding evaluation for omnimodal LLMs. *arXiv preprint arXiv:2603.19217*, 2026b.

Zining Wang, Zhihang Yuan, Yingjie Zhai, Wenshuo Li, Han Shu, Ruihao Gong, Jinyang Guo, and Xianglong Liu. OmniFit: Bridging modalities via layer-adaptive token compression for omnimodal large language models. In *Forty-third International Conference on Machine Learning*, 2026.

Zhifei Xie and Changqiao Wu. Mini-Omni2: Towards open-source GPT-4o with vision, speech and duplex capabilities. *arXiv preprint arXiv:2410.11190*, 2024.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-Omni technical report. *arXiv preprint arXiv:2503.20215*, 2025a.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. Qwen3-Omni technical report. *arXiv preprint arXiv:2509.17765*, 2025b.

Morunliu Yang, Ruotao Xu, Le Li, Yue Wang, Jianxin Zhang, Juntao Li, Yihang Lou, Siwei Feng, and Peifeng Li. OmniSelect: Dynamic modality-aware token compression for efficient omnimodal large language models. *arXiv preprint arXiv:2605.18041*, 2026.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802. IEEE, 2025.

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*, pp. 881–916, 2025a.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis A Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. SparseVLM: Visual token sparsification for efficient vision-language model inference. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 74840–74857, 2025b.

Yiming Zhong, Chang Nie, and Caifeng Shan. Omni-Prune: Query-aware unified token pruning for efficient omnimodal large language models. *arXiv preprint arXiv:2607.23445*, 2026.

Jiedong Zhuang, Lu Lu, Ming Dai, Rui Hu, Jian Chen, Qiang Liu, and Haoji Hu. ST3: Accelerating multimodal large language model by spatial-temporal visual token trimming. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11049–11057, 2025.

A APPENDIX

In this part, we provide additional details and more experimental results on our approach. The supplementary material is structured as follows:

- § $A.1$: Experimental setup;
- § $A.2$: Method implementation details;
- § $A.3$: Supplementary results and analysis;
- § $A.4$: Qualitative examples;
- § $A.5$: Limitations and future work.

A.1 EXPERIMENTAL SETUP

A.1.1 BENCHMARKS AND EVALUATION PROTOCOL

We evaluate audio-visual understanding on six benchmarks covering different video durations, modality dependencies, and reasoning requirements. We use the official or paper-specified split for each benchmark. The compressed and uncompressed models receive the same video, audio track, prompt, decoding settings, and answer parser. The only changed input is the audio-visual token sequence passed to the language model.

- **ShortVid-Bench.** ShortVid-Bench evaluates fast-paced real-world short videos. The benchmark tests whether compression preserves brief visual and audio events instead of allocating tokens uniformly across the clip. The short-video setting directly evaluates the local budget allocator.
- **WorldSense.** WorldSense covers diverse real-world scenarios and emphasizes synchronized audio-visual understanding. The benchmark tests whether the model combines information from both modalities when relevant content occurs at related times. This focus makes WorldSense useful for evaluating the Correlation-Aware Selection policy.
- **OmniVideoBench.** OmniVideoBench evaluates reasoning over complementary audio and visual information. Some questions require information from both modalities. The benchmark tests whether separate selection preserves modality-specific information.
- **Video-Holmes.** Video-Holmes focuses on complex reasoning over clues distributed throughout a video. The task evaluates both temporal budget allocation and the preservation of supporting context after token removal. Since clues can occur at different times, the benchmark also tests whether the allocator retains enough context to connect them.
- **JointAVBench.** JointAVBench requires both modalities across different audio types and temporal spans. The benchmark tests whether the compression policy adapts when audio and video are complementary in some examples and overlapping in others.
- **LVOmniBench.** LVOmniBench evaluates audio-visual understanding over long videos. Its longer sequences make token allocation more challenging and test whether the method reduces redundancy without losing information from distant events. The long-video setting stresses the distribution of a finite global budget across many temporal chunks.
- **Evaluation protocol.** We use the same input, prompt, generation settings, and answer parser for all methods. We report the official metric for each benchmark and its task categories. For efficiency, we measure prefill time, time to first token, decoding throughput, end-to-end latency, and peak memory under the same hardware and decoding settings. Generation length is fixed across methods.

A.1.2 COMPARISON BASELINES

We compare OmniTrim with four training-free token compression methods. The baselines cover video pruning applied to both modalities, audio-guided visual pruning, and independent audio/video compression. All methods use the same prompts, model checkpoint, generation settings, answer parser, and hardware conditions. We evaluate each baseline at the requested retention ratios while preserving its original selection rule.

- **FastVID.** FastVID partitions a video into temporally ordered segments and performs density-based token pruning to preserve spatial and temporal information. We apply it to both audio and visual modalities using its original selection settings. FastVID serves as a density-based comparison without OmniTrim's adaptive chunk budget or cross-modal selection switch.
- **DyCoke.** DyCoke combines inter-frame token merging with dynamic key-value-cache pruning. We apply its video compression procedure to both audio and visual modalities for a matched V&A comparison, without adding a training stage. Unlike OmniTrim, DyCoke uses inter-frame redundancy and cache pruning rather than a chunk-level audio-visual correlation estimate.
- **OmniZip.** OmniZip uses audio importance to guide visual token pruning. The baseline tests whether audio-guided visual pruning is sufficient when the two modalities are complementary or redundant. We retain its audio-guided scoring rule and do not add OmniTrim's temporal budget redistribution or joint selection.
- **OmniScope.** OmniScope compresses audio and video independently. The baseline provides a comparison to modality-wise pruning without a chunk-level decision between separate and joint selection. The comparison isolates the effect of estimating the audio-visual correlation within each temporal chunk.

A.2 Method Implementation Details

We detail the omni-modal input representation, budget smoothing after Information-Aware Budgeting, and token merging in joint selection.

A.2.1 Omni-Modal Input Representation

An omni-modal model processes audio, video, and text in one sequence. The audio and visual encoders convert the two modalities into token sequences, which are then processed together with the text tokens by the language model. Let N_v and N_a denote the numbers of visual and audio tokens before compression. For long videos, audio-visual tokens can make up most of the input sequence. OmniTrim compresses the audio-visual tokens and keeps the text tokens unchanged.

In our experiments, Qwen2.5-Omni receives synchronized audio and video together with a textual prompt. The multimodal encoders map the audio and visual modalities into the language-model embedding space, after which the model reasons over the combined sequence. The number of audio-visual tokens grows with the video duration, while the text input remains comparatively short. Compression is applied before the language-model forward pass, with the original temporal order and text prompt unchanged.

A.2.2 Budget Smoothing

Information-Aware Budgeting assigns larger retention ratios to informative chunks, but abrupt changes between adjacent chunks can produce uneven temporal coverage. We smooth the local ratios before converting them into token budgets. Let $u_k = \mathrm{clip}(\widehat{r}_k, r_{\min}, r_{\max})$ denote the bounded ratio from Equation 3 for chunk k. We use $r_{\min} = 0.05$ and $r_{\max} = 0.98$. We first compute forward and backward exponential averages:

$$\begin{aligned} f_1 &= u_1, \quad f_k = s u_k + (1-s) f_{k-1}, \quad k = 2, \dots, K, \\ b_K &= u_K, \quad b_k = s u_k + (1-s) b_{k+1}, \quad k = K-1, \dots, 1. \end{aligned} \tag{7}$$

Here, $s \in [0, 1]$ is the smoothing coefficient, distinct from the activity weight α. We average the two directions and clip the result:

$$q_k = \mathrm{clip}\left(\frac{f_k + b_k}{2}, r_{\min}, r_{\max}\right). \tag{8}$$

Using both directions avoids privileging the beginning or end of the sequence. For a single chunk, the bounded ratio is returned unchanged.

We further limit the change between neighboring ratios to δ. In a forward pass, q_k is clipped to $[q_{k-1} - \delta, q_{k-1} + \delta]$; in a backward pass, it is clipped to $[q_{k+1} - \delta, q_{k+1} + \delta]$. Each update is also

clipped to $[r_{\min}, r_{\max}]$. The implementation performs two forward–backward clipping rounds. The parameter-sweep configuration uses $s = 0.30$ and $\delta = 0.08$.

The smoothed ratios determine the preliminary chunk budgets. Smoothing does not itself enforce the global token count: rounding and selection constraints can change the sum of retained tokens. The subsequent budget reconciliation restores the target B by removing the lowest-scoring retained tokens or restoring the highest-scoring discarded tokens as needed. Thus, temporal smoothing controls local allocation changes while reconciliation enforces the shared budget.

A.2.3 Token Merging in Joint Selection

For a chunk with $\rho > \tau$, joint selection retains a compact set of representatives from the audio-visual tokens. Simply discarding the remaining tokens can lose information that is related to these representatives. We therefore aggregate the discarded features into the retained tokens using similarity-weighted residual merging. This operation updates the retained embeddings without adding tokens to the chunk budget.

Assignment to retained representatives. Let $\mathcal{S}$ and $\mathcal{D}$ denote the selected and discarded token indices in the current chunk, respectively, and let z_i denote an audio or visual token in the shared embedding space. Each discarded token is assigned to its most similar selected representative:

$$j^\star(i) = \arg\max_{j \in \mathcal{S}} \cos(z_i, z_j), \qquad \mathcal{D}_j = \{i \in \mathcal{D} : j^\star(i) = j\}. \tag{9}$$

The assignment considers the joint audio-visual pool within the chunk, so a discarded token can contribute to a retained representative from either modality. Similarities are computed from the embeddings before merging.

Similarity-weighted merging. For a representative with $\mathcal{D}_j \neq \emptyset$, we normalize the cosine similarities over its assigned discarded tokens:

$$w_{ij} = \frac{\exp(\cos(z_i, z_j))}{\sum_{\ell \in \mathcal{D}_j} \exp(\cos(z_\ell, z_j))}, \qquad i \in \mathcal{D}_j. \tag{10}$$

More similar tokens receive larger weights. The retained embedding is then updated by

$$\widehat{z}_j = \frac{z_j + \beta_k \sum_{i \in \mathcal{D}_j} w_{ij} z_i}{1 + \beta_k \sum_{i \in \mathcal{D}_j} w_{ij}}. \tag{11}$$

Here, $\beta_k \geq 0$ controls the contribution of discarded features in chunk k. In the joint branch, the implementation uses the base merging strength for $\tau < \rho < \max(0.75, \tau)$ and increases it by a factor of 1.4 when $\rho \geq \max(0.75, \tau)$. Since the weights sum to one for each non-empty $\mathcal{D}_j$, the denominator is $1 + \beta_k$, keeping the update a normalized combination of the original representative and its aggregated features. If $\mathcal{D}_j$ is empty, we leave z_j unchanged. The updated embeddings occupy the original selected positions, preserving the selected token count, temporal order, and text tokens.

A.3 Supplementary Results and Analysis

A.3.1 Computational Cost

We analyze the computation of the language-model backbone and the additional cost of token compression. Let n be the input token count, d the hidden dimension, d_{ff} the FFN intermediate dimension, T the number of transformer layers, and R the number of generated tokens. For one transformer layer, the prefill FLOPs can be approximated by

$$\mathcal{F}_{\text{pre}}(n) = 4nd^2 + 2n^2 d + 2ndd_{\text{ff}}. \tag{12}$$

The first and third terms correspond to the linear projections and the FFN, while the quadratic term comes from self-attention. During decoding, the KV cache changes the attention cost as the sequence grows. For R generated tokens, the corresponding approximation is

$$\mathcal{F}_{\text{dec}}(n, R) = R(4d^2 + 2dd_{\text{ff}}) + 2d \sum_{i=1}^{R} (n + i). \tag{13}$$

Table 4: **Parameter sensitivity results.** Accuracy (%) on WorldSense and Video-Holmes using Qwen2.5-Omni-7B. Bold parameter values mark the settings stated in the main text.

α			λ			γ			τ			η		
Value	**45%**	**35%**	**Value**	**45%**	**35%**	**Value**	**45%**	**35%**	**Value**	**45%**	**35%**	**Value**	**45%**	**35%**
WorldSense														
0.35	46.1	45.0	0.15	45.9	44.6	0.0	46.3	44.9	0.58	45.7	44.5	0	46.1	44.6
0.50	46.2	45.1	0.35	46.1	45.0	0.4	46.2	44.7	0.62	46.2	44.5	0.125	46.1	44.8
0.65	**46.5**	**45.7**	**0.55**	**46.5**	**45.7**	**0.8**	**46.5**	**45.7**	**0.66**	**46.5**	**45.7**	**0.25**	**46.5**	**45.7**
0.80	45.9	44.7	0.75	46.0	44.7	1.2	45.9	44.4	0.70	46.5	45.7	0.375	46.3	44.8
0.95	46.3	44.9	0.95	46.2	44.9	1.6	46.2	44.8	0.74	46.3	45.2	0.5	46.3	45.0
Video-Holmes														
0.35	46.7	45.3	0.15	46.2	45.4	0.0	46.3	45.4	0.58	45.4	44.8	0	46.0	45.3
0.50	46.5	45.6	0.35	46.8	45.9	0.4	46.5	45.4	0.62	45.2	44.5	0.125	46.4	45.1
0.65	**46.3**	**46.4**	**0.55**	**46.3**	**46.4**	**0.8**	**46.3**	**46.4**	**0.66**	**46.3**	**46.4**	**0.25**	**46.3**	**46.4**
0.80	46.2	45.4	0.75	46.3	45.3	1.2	46.4	45.7	0.70	46.3	46.4	0.375	45.9	45.3
0.95	46.3	45.3	0.95	46.3	45.6	1.6	46.2	44.9	0.74	46.3	45.6	0.5	46.1	45.3

The total backbone cost is $T[\mathcal{F}_{\text{pre}}(n)+\mathcal{F}_{\text{dec}}(n, R)]$. This estimate covers the language-model backbone; the audio and visual encoder costs are unchanged between the compressed and uncompressed inputs.

Let L_t be the number of text tokens, $N = N_v + N_a$ the number of audio-visual tokens before compression, and B the number retained by OmniTrim. The uncompressed and compressed input lengths are $n = L_t + N$ and $n' = L_t + B$, respectively. The prefill cost changes from $\mathcal{F}_{\text{pre}}(n)$ to $\mathcal{F}_{\text{pre}}(n')$, while decoding changes according to $\mathcal{F}_{\text{dec}}(n, R)$ and $\mathcal{F}_{\text{dec}}(n', R)$. Longer audio-visual inputs benefit more from compression because self-attention has a quadratic term.

The compression stage adds a preprocessing cost. Computing token norms, chunk means, and local similarities takes $\mathcal{O}(Nd + Kd)$ time for K chunks. Full token ranking costs $\mathcal{O}(N \log N)$, while a partial top-B operation costs $\mathcal{O}(N \log B)$. If every discarded token is compared with all selected tokens for token merging, the worst-case cost is $\mathcal{O}(NBd)$. Budgeting and selection run once before the language-model forward pass.

A.3.2 Parameter Sensitivity

Figure 4 reports the sensitivity to α, λ, γ, and η on WorldSense and Video-Holmes. Table 4 provides the corresponding numerical results and additionally reports the sensitivity to τ. These experiments use Qwen2.5-Omni-7B and follow the evaluation protocol of the main experiments. The results show how the weighting and routing parameters affect performance at the two retention settings. At 35% retention, the reported scores for α and λ are highest at the default settings on both datasets, whereas the best-performing values differ across datasets at 45% retention. For γ, increasing the budget adjustment strength beyond 0.8 does not consistently improve accuracy. At the 35% retention setting reported in the table, $\tau = 0.70$ yields 45.7% on WorldSense and 46.4% on Video-Holmes; increasing τ to 0.74 reduces these scores to 45.2% and 45.6%, respectively. For η, the reported scores are highest at 0.25 on both datasets at 35% retention, with lower scores at all other tested values.

A.3.3 Effect of the Maximum Number of Input Frames

Table 5 compares OmniTrim, OmniZip, and OmniScope on selected reasoning subtasks from Video-Holmes and WorldSense with maximum input frame counts of 128, 256, 512, and the default 768 using Qwen2.5-Omni-7B. OmniTrim achieves the highest average accuracy across the six subtasks at all four frame limits. Its average score rises from 45.9% at 128 frames to 47.3% at 512 frames

Table 5: **Effect of the maximum number of input frames on selected reasoning subtasks.** Accuracy (%) using Qwen2.5-Omni-7B at 25% token retention; 768 frames is the default limit. For Video-Holmes, SR, MHR, and TCI denote Social Reasoning, Multimodal Hint Reasoning, and Temporal Causal Inference, respectively. For WorldSense, CR, TP, and SR denote Causal Reasoning, Temporal Prediction, and Spatial Relation, respectively. Avg. is the mean of the six displayed subtasks; best scores per frame limit are bold.

Frames	Method	Video-Holmes			WorldSense			Avg.
		SR	MHR	TCI	CR	TP	SR	
128	OmniZip	59.2	**38.9**	**38.1**	49.0	45.5	28.9	43.3
	OmniScope	59.2	38.0	35.5	55.6	**50.9**	28.4	44.6
	OmniTrim (Ours)	**59.6**	38.3	36.3	**57.6**	**50.9**	**32.5**	**45.9**
256	OmniZip	61.0	38.3	**38.5**	47.0	43.6	27.4	42.6
	OmniScope	59.6	**38.9**	37.4	53.6	48.2	**31.5**	44.9
	OmniTrim (Ours)	**62.7**	**38.9**	37.0	**57.6**	**50.0**	**31.5**	**46.3**
512	OmniZip	59.2	38.9	38.1	47.0	44.5	27.9	42.6
	OmniScope	58.9	39.5	37.4	53.6	45.5	30.5	44.2
	OmniTrim (Ours)	**62.0**	**40.7**	**39.2**	**59.6**	**50.0**	**32.5**	**47.3**
768	OmniZip	58.2	39.2	38.5	47.0	45.5	24.9	42.2
	OmniScope	59.6	38.9	37.7	53.6	45.5	29.9	44.2
	OmniTrim (Ours)	**63.0**	**40.7**	**39.2**	**59.6**	**50.0**	**31.0**	**47.2**

and remains similar at 47.2% with 768 frames. At 512 and 768 frames, OmniTrim leads on all six displayed subtasks. These results show that its advantage extends across the tested input lengths, with most of the average improvement occurring by 512 frames.

A.4 Qualitative Examples

Figure 6 presents four question-answering examples with sampled video frames, audio descriptions, and predictions from OmniZip, OmniScope, and OmniTrim. The questions ask about a person's relative position, the reason for installing lights, the equipment used to film a video, and the meaning of an action in a scene. In all four examples, the answers from OmniTrim are marked as correct, whereas those from OmniZip and OmniScope are marked as incorrect.

A.5 Limitations and Future Work

OmniTrim estimates audio-visual correlation using mean-pooled chunk representations. While computationally efficient, this coarse representation may overlook fine-grained correspondences or brief temporal offsets between audio and video, limiting the reliability of the local selection decision. Fixed chunk boundaries can also split a short event across adjacent chunks, making its information content harder to capture with individual chunk scores. Future work could address these limitations through finer-grained audio-visual alignment and content-adaptive temporal partitioning, while preserving the low overhead of training-free token compression.



Audio: *Instrumental ensemble music plays continuously. No clear speech is audible in the selected segment.*

Question: *From the audience's perspective, where is the white-haired elderly man positioned relative to the man wearing a gray long-sleeved shirt?*

OmniZip: A. Behind the man wearing a gray long-sleeved shirt. ✗

OmniScope: A. Behind the man wearing a gray long-sleeved shirt. ✗

OmniTrim: C. To the left of the man wearing a gray long-sleeved shirt. ✓



Audio: *A woman explains how she set up her PS5 gaming station and added RGB lighting. She discusses buying Nanoleaf lights, liking their appearance, and assembling them with a friend's help.*

Question: *Why did the woman install Nanoleaf light strips?*

OmniZip: B. To decorate the room. ✗ **OmniScope:** B. To decorate the room. ✗

OmniTrim: C. To create a gaming station. ✓



Audio: *A man describes a fully manual M42 screw-mount lens and its adapter for a Canon FD mount. He explains that he took the lens on a Thanksgiving trip to Paso Robles to shoot photographs.*

Question: *What did the man use to film the Thanksgiving video?*

OmniZip: C. Canon camera. ✗ **OmniScope:** C. Canon camera. ✗

OmniTrim: B. M42-threaded lens. ✓



Audio: *Background music accompanies faint tapping and contact sounds consistent with tableware being handled. No clear dialogue is identified in this segment.*

Question: *What is the metaphor of the man cutting steak at 1:03?*

OmniZip: D. Symbolize the fragmentation of time. ✗

OmniScope: D. Symbolize the fragmentation of time. ✗

OmniTrim: A. Implied homicide method. ✓

Figure 6: **Qualitative question-answering examples.** Each example shows sampled video frames, an audio description, the question, and predictions from OmniZip, OmniScope, and OmniTrim. Green and red denote correct and incorrect answers, respectively.