TIMEPRUNE: ADAPTIVE TEMPORAL RESOLUTION FOR EFFICIENT AUDIO LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Audio representations form temporally ordered token trajectories whose variation can differ substantially over time. Processing dense audio tokens throughout Audio Large Language Models (ALLMs), however, incurs substantial computational overhead. Existing audio token compression methods address this inefficiency by exploiting token importance, redundancy, or temporal structure, but typically do not directly assess how faithfully the retained tokens preserve the underlying representation trajectory. We introduce **TimePrune**, a training-free framework that preserves temporal representation trajectories by adaptively determining temporal resolution under a given token budget. At its core, Time-Aware RDP recursively refines intervals whose intermediate representations are insufficiently described by their temporal anchors, yielding adaptive local anchor density according to trajectory deviation. We further introduce Temporal Coverage Refinement to regulate excessively sparse temporal intervals while preserving trajectory-driven refinement. Extensive experiments across four ALLMs and five audio understanding benchmarks demonstrate the effectiveness of TimePrune across different models and token budgets. In our efficiency analysis, TimePrune retains only **15%** of audio tokens while preserving **98.4%** of the Vanilla score and reducing audio-token-dependent LLM computation by **85.2%**, demonstrating its effectiveness for efficient ALLM inference.

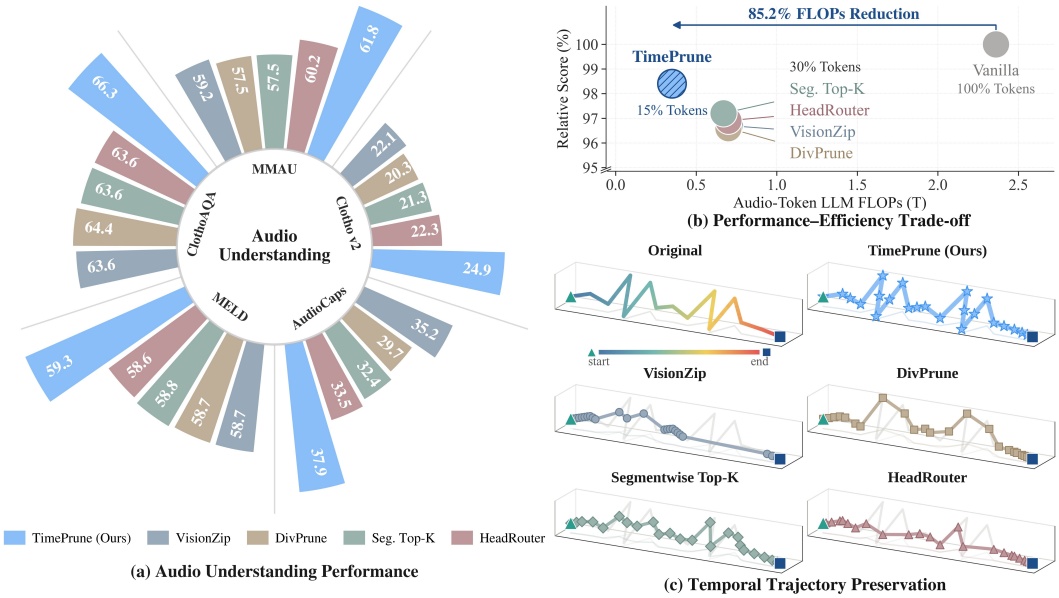


Figure 1: **Performance and efficiency of TimePrune.** (a) Audio understanding performance. (b) TimePrune preserves 98.4% of the Vanilla score with 15% of audio tokens while reducing audio-token-dependent LLM FLOPs by 85.2%. (c) Temporal trajectory preservation with retained anchors.

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.

1 INTRODUCTION

Recent multimodal large language models have demonstrated strong audio understanding capabilities across speech, environmental sounds, and music Shao et al. (2026). In this work, we refer to models capable of processing audio inputs as Audio Large Language Models (ALLMs), encompassing both audio-focused and omni-modal architectures such as AudioFlamingo Next Ghosh et al. (2026), MOSS-Audio Yang et al. (2026), Qwen2.5-Omni Xu et al. (2025a), and Qwen3-Omni Xu et al. (2025b). These models transform continuous audio signals into temporally ordered representations that are subsequently processed by the language model. Preserving fine-grained acoustic information, however, often produces hundreds or thousands of audio tokens, resulting in substantial computational and memory overhead as input duration increases. Efficiently reducing these dense audio representations while preserving information required for downstream understanding is therefore important for scalable ALLM inference.

Motivation. Existing studies improve token efficiency through pruning and merging using different sources of information. Audio token compression exploits attention- and task-based importance, redundancy, and temporal structure Lin et al. (2025); Gibier et al. (2026); He et al. (2026); Luo et al. (2026). Visual token compression also considers attention-based saliency, diversity, and representativeness Chen et al. (2024); Yang et al. (2025); Alvar et al. (2025); Cho et al. (2026b); Xu et al. (2026). Multimodal token compression further leverages cross-modal correspondence and multimodal interactions to guide token reduction Tao et al. (2026); Ding et al. (2026); Wang et al. (2026). While these approaches provide effective criteria for selecting or compressing tokens, they do not directly assess how faithfully the retained representations preserve the underlying temporal trajectory. This distinction is particularly important for audio, where representations form temporally ordered trajectories whose structure evolves over time. We therefore revisit **audio token pruning** from a **temporal trajectory preservation** perspective by jointly considering trajectory fidelity and temporal coverage under limited token budgets.

Key Observations. From this perspective, we first analyze existing pruning strategies in terms of trajectory fidelity and temporal coverage, which characterize how faithfully the retained tokens approximate the original representation trajectory and whether they avoid excessive temporal sparsity, respectively. We observe that these two properties vary substantially across pruning strategies even under the same token budget, and that trajectory-driven refinement can achieve high trajectory fidelity while producing excessively sparse temporal intervals. We further find that trajectory-driven refinement based on time-synchronized trajectory deviation naturally induces non-uniform temporal resolution without explicitly using local representation variation for token allocation. Together, these observations motivate trajectory-driven adaptive resolution while regulating excessive temporal sparsity.

TimePrune. Based on these observations, we introduce TimePrune, a training-free audio token pruning framework that adaptively determines temporal resolution under a limited token budget. At its core, **Time-Aware RDP** extends the Ramer–Douglas–Peucker (RDP) trajectory simplification principle Ramer (1972); Douglas & Peucker (1973) by measuring representation deviation through time-synchronized interpolation Meratnia & de By (2004); Tamilmani & Stefanakis (2019). It recursively selects temporal anchors where intermediate representations are insufficiently approximated by their current anchors, producing adaptive local anchor density according to the representation trajectory. We further introduce **Temporal Coverage Refinement** to prioritize excessively sparse temporal intervals, complementing trajectory-driven anchor selection by regulating excessive temporal sparsity. The resulting anchors directly constitute the compressed audio token sequence without requiring additional training, token scoring, or merging.

Results. As summarized in Fig. 1, extensive experiments across multiple ALLMs and diverse audio understanding benchmarks demonstrate the effectiveness of TimePrune across different model architectures, tasks, and compression budgets. In our efficiency analysis on Qwen2.5-Omni-7B, TimePrune retains only **15%** of audio tokens while preserving **98.4%** of the Vanilla score and reducing audio-token-dependent LLM computation by **85.2%**.

Our main contributions are summarized as follows:

- We revisit audio token pruning from a **temporal trajectory preservation** perspective by jointly examining trajectory fidelity and temporal coverage under limited token budgets.

- We propose **TimePrune**, a training-free audio token pruning framework that adaptively determines temporal resolution through Time-Aware RDP while regulating excessive temporal sparsity through Temporal Coverage Refinement.
- We validate TimePrune across diverse ALLMs, audio understanding tasks, and compression budgets, demonstrating consistent effectiveness under aggressive token compression while substantially reducing audio-token-dependent LLM computation.

2 Motivating Observations

2.1 Temporal Trajectory Preservation

Observation 1: Temporal Trajectory Preservation Varies across Compression Methods. We compare representative compression methods under the same token budget, including Uniform retention as a temporal-spacing reference, VisionZip Yang et al. (2025), DivPrune Alvar et al. (2025), Segmentwise Top-K Gibier et al. (2026), HeadRouter He et al. (2026), Time-Aware RDP as a trajectory-driven reference, and TimePrune. We evaluate them on Clotho v2 Drossos et al. (2020), MELD Poria et al. (2019), and MMAU Sakshi et al. (2025). Viewing retained tokens as temporal anchors, we characterize them by **trajectory fidelity** and **temporal coverage**. We measure trajectory fidelity using Synchronous Euclidean Distance (SED) Meratnia & de By (2004); Lin et al. (2019), which measures time-synchronized deviation from the original trajectory.

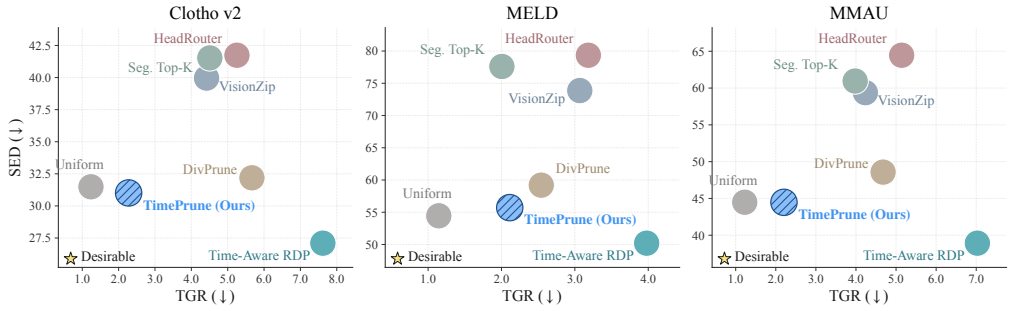


Figure 2: **Trajectory fidelity and temporal coverage across compression methods** at 75% pruning. Lower SED indicates higher trajectory fidelity, while lower TGR indicates fewer excessively sparse temporal intervals. Uniform has TGR near 1 by construction.

For temporal coverage, we introduce the **Temporal Gap Ratio (TGR)**, which normalizes the largest inter-anchor gap by the nominal uniform spacing:

$$\mathrm{TGR} = \frac{\max_{1 \le i < K}(a_{i+1} - a_i)}{(T-1)/(K-1)}, \tag{1}$$

where a_i is the temporal index of the i-th retained anchor, T is the sequence length, and K is the number of retained anchors.

As shown in Fig. 2, the methods occupy distinct regions of the fidelity–coverage space despite retaining the same number of tokens. Time-Aware RDP achieves the lowest SED but exhibits substantially larger TGR, whereas Uniform maintains near-ideal TGR but higher approximation error. TimePrune substantially reduces the TGR of Time-Aware RDP while maintaining relatively low approximation error. These results motivate jointly considering trajectory fidelity and temporal coverage. Additional analyses across compression methods and pruning ratios are provided in Appendix D.

2.2 Adaptive Temporal Resolution

Observation 2: Trajectory-driven refinement naturally induces adaptive temporal resolution. Observation 1 shows that Time-Aware RDP achieves high trajectory fidelity but can produce excessively sparse temporal intervals. We next examine whether its trajectory-driven refinement naturally adapts temporal resolution to local representation structure. Specifically, Time-Aware RDP

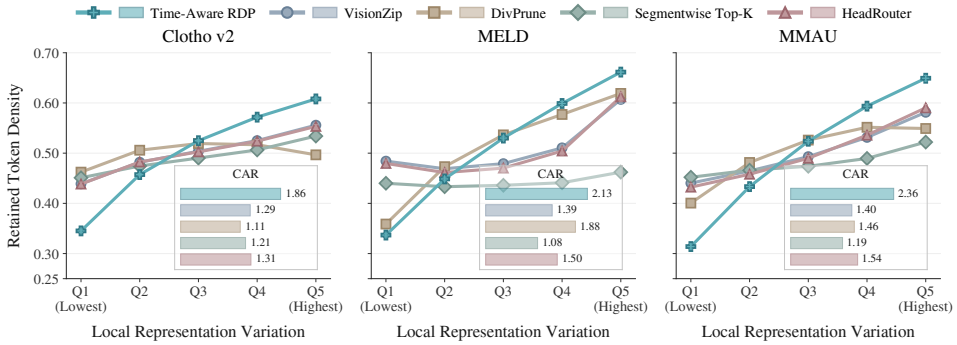


Figure 3: **Token allocation across local representation variation.** Retained token density across local representation-variation quantiles (Q1–Q5) on Clotho v2, MELD, and MMAU under 50% pruning. $\mathrm{CAR} = D_{Q5}/D_{Q1}$ measures the relative allocation between the highest- and lowest-variation regions. Local representation variation is not used for token selection by Time-Aware RDP.

recursively selects temporal anchors based on time-synchronized representation deviation, without explicitly using local representation variation as a selection criterion.

Adaptation to Local Representation Structure. We divide each audio sequence into equal temporal regions and quantify the local representation variation of each region $\mathcal{R}_m$ as

$$C_m = \frac{1}{|\mathcal{R}_m| - 1} \sum_{\substack{t \in \mathcal{R}_m \\ t+1 \in \mathcal{R}_m}} \left(1 - \cos(\mathbf{x}_t, \mathbf{x}_{t+1})\right). \tag{2}$$

We group the regions into five variation quantiles (Q1–Q5) and measure their retained token density. We further define the **Complexity Allocation Ratio (CAR)** as $\mathrm{CAR} = D_{Q5}/D_{Q1}$, where D_{Q5} and D_{Q1} denote the retained token densities in the highest- and lowest-variation quantiles, respectively. As shown in Fig. 3, Time-Aware RDP exhibits a pronounced increase in retained density toward higher-variation regions and achieves the highest CAR across all three datasets. Despite not explicitly using local representation variation for token selection, Time-Aware RDP therefore allocates relatively denser temporal anchors to higher-variation regions. This suggests that trajectory-driven refinement naturally induces adaptive temporal resolution according to the underlying representation structure.

Qualitative Temporal Allocation. Fig. 4 further illustrates the non-uniform temporal allocation induced by Time-Aware RDP within an audio sample, with substantially different local anchor densities across temporal regions. Together with the aggregate relationship between representation variation and retained density in Fig. 3, this further supports that trajectory-driven refinement naturally induces adaptive temporal resolution. Combined with Observation 2.1, these findings motivate preserving this adaptive behavior while regulating excessive temporal sparsity. Additional analyses are provided in Appendix E.

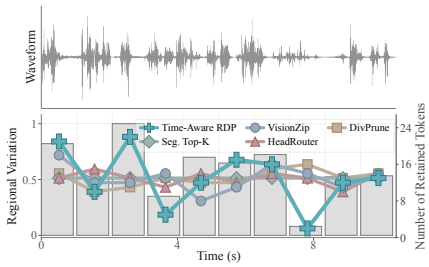


Figure 4: **Adaptive temporal allocation.** Regional representation variation and retained token allocation over time.

3 Methodology

3.1 Overview

We propose **TimePrune**, a training-free audio token pruning framework designed to preserve temporal representation trajectories under a fixed token budget, as illustrated in Fig. 5. Given a temporally ordered audio representation sequence $\mathbf{X} = \{\mathbf{x}_t\}_{t=1}^{T}$ of T audio tokens and a target budget K

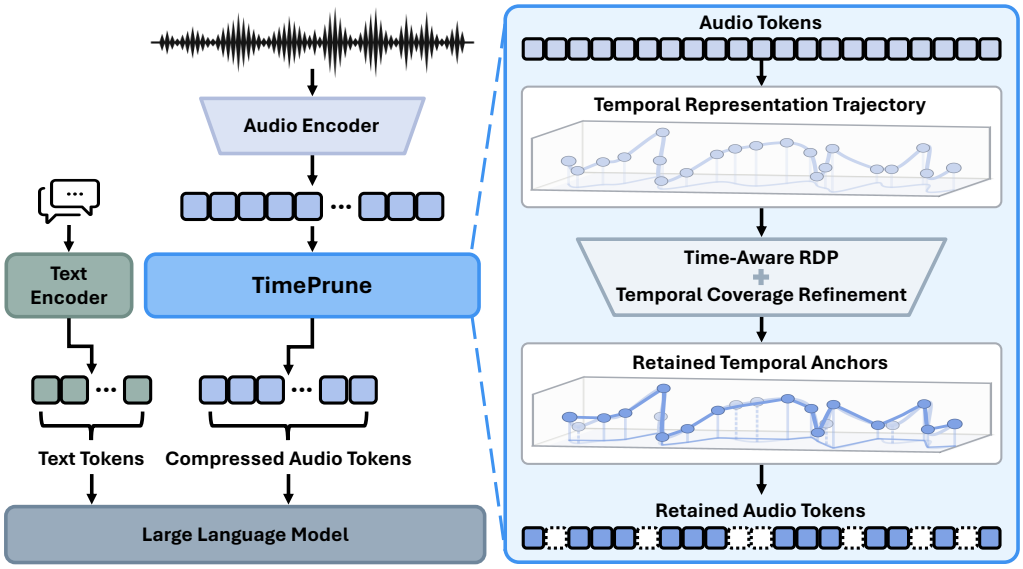


Figure 5: **Overview of TimePrune.** TimePrune combines Time-Aware RDP for trajectory-driven adaptive temporal resolution with Temporal Coverage Refinement for regulating excessive temporal sparsity under a limited token budget.

specifying the number of tokens to retain, TimePrune constructs an ordered set of temporal anchor indices $\mathcal{A} = \{a_1, \ldots, a_K\}$, where each a_k denotes the index of a retained audio token. Time-Aware RDP recursively refines the representation trajectory by inserting additional anchors where the current anchors are insufficient to faithfully describe the intermediate representations, thereby inducing adaptive temporal resolution along the sequence. Temporal Coverage Refinement complements this trajectory-driven refinement by prioritizing excessively sparse temporal intervals for refinement. Together, these components preserve the temporal representation trajectory by jointly considering trajectory fidelity and temporal coverage within the target budget.

3.2 Time-Aware RDP

Time-Synchronized Representation Deviation. We regard the temporally ordered audio representations as a trajectory in the representation space. Time-Aware RDP builds on the Ramer–Douglas–Peucker (RDP) trajectory simplification principle Ramer (1972); Douglas & Peucker (1973). To measure representation deviation along the temporal trajectory, we adopt time-synchronized interpolation Meratnia & de By (2004); Tamilmani & Stefanakis (2019). For an interval bounded by two neighboring temporal anchor indices $l < r$, we approximate each intermediate representation $\mathbf{x}_t$, where $l < t < r$, according to its relative temporal position as

$$\tau_t = \frac{t-l}{r-l}, \qquad \hat{\mathbf{x}}_t = (1-\tau_t)\mathbf{x}_l + \tau_t \mathbf{x}_r. \tag{3}$$

We then measure its deviation from the original representation as $D_t = d(\mathbf{x}_t, \hat{\mathbf{x}}_t)$, where $d(\cdot,\cdot)$ denotes a representation deviation metric. TimePrune supports different deviation measures, including ℓ_1 distance, ℓ_2 distance, cosine distance $(1 - \cos(\cdot,\cdot))$, and negative dot product, with larger values consistently indicating greater representation deviation.

Recursive Temporal Refinement. For each interval $[l, r]$ between neighboring temporal anchors, we identify the intermediate representation with the largest deviation,

$$t^*_{l,r} = \arg\max_{l<t<r} D_t, \qquad D^*_{l,r} = D_{t^*_{l,r}}. \tag{4}$$

Starting from the initial endpoint anchor set $\mathcal{A} = \{1, T\}$, Time-Aware RDP repeatedly selects the interval with the largest $D^*_{l,r}$ and inserts $t^*_{l,r}$ into $\mathcal{A}$ as a new temporal anchor. The resulting subintervals are recursively evaluated until the target budget K is reached. Intervals that are sufficiently described by their current anchors undergo fewer refinements and remain at coarser temporal resolution, whereas intervals with relatively larger trajectory deviation are preferentially refined with additional anchors, resulting in finer temporal resolution. Time-Aware RDP therefore induces adaptive temporal resolution directly from the representation trajectory without relying on predefined temporal partitions or explicit local-complexity-based allocation.

3.3 Temporal Coverage Refinement

Although Time-Aware RDP adaptively refines the trajectory based on representation deviation, deviation-driven refinement alone may concentrate anchors within limited portions of the sequence and leave excessively large temporal gaps elsewhere. Motivated by the fidelity–coverage behavior observed in Sec. 2.1, we complement trajectory-driven refinement with Temporal Coverage Refinement. Given the sequence length T and target budget K, we define the nominal anchor spacing $\bar{g}$ and the coverage threshold $g_{\max}$ as

$$\bar{g} = \frac{T-1}{K-1}, \qquad g_{\max} = \beta \bar{g}, \tag{5}$$

where $\beta \geq 1$ controls the tolerance for temporal gaps. Smaller β triggers coverage refinement at shorter gaps, whereas $\beta \to \infty$ recovers Time-Aware RDP without coverage refinement.

An interval between neighboring anchors l and r violates the coverage criterion when $r - l > g_{\max}$. If coverage-violating intervals exist, TimePrune prioritizes their refinement; otherwise, it selects the interval with the largest representation deviation $D^*_{l,r}$. Once an interval $[l, r]$ is selected, the intermediate representation with the largest time-synchronized representation deviation is retained as the new temporal anchor,

$$t^*_{l,r} = \arg\max_{l<t<r} D_t. \tag{6}$$

Thus, Temporal Coverage Refinement determines which excessively sparse interval should be prioritized, while Time-Aware RDP determines which representation within the selected interval should be retained. This preserves trajectory-driven anchor selection while regulating excessive temporal gaps, allowing TimePrune to jointly consider trajectory fidelity and temporal coverage.

The refinement process terminates once K temporal anchors are selected. Given the final ordered anchor set $\mathcal{A}$ with $a_1 < \cdots < a_K$, the compressed audio representation is constructed as

$$\mathbf{X}_{\text{TimePrune}} = \{\mathbf{x}_{a_i}\}_{i=1}^{K}. \tag{7}$$

The selected temporal anchors directly constitute the compressed audio sequence without additional token scoring, clustering, merging, or model training, enabling direct application to pretrained ALLMs.

4 Experiments

In this section, we first describe the experimental settings (Sec. 4.1), followed by the main results under different token budgets and across different ALLMs (Sec. 4.2). We then present ablation studies on trajectory-driven refinement, temporal coverage refinement, and representation deviation metrics (Sec. 4.3), followed by an efficiency analysis (Sec. 4.4). Additional experiments and analyses are provided in the Appendix.

4.1 Experimental Settings

Models. We evaluate TimePrune on four ALLMs with different audio representation pipelines: Qwen2.5-Omni-7B (Xu et al., 2025a), AudioFlamingo-Next-Instruct (Ghosh et al., 2026), Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025b), and MOSS-Audio-8B-Instruct (Yang et al., 2026). The first two use Whisper-based audio encoders (Radford et al., 2023), while the latter two employ distinct Transformer-based audio encoders. For omni-modal models, we use audio-only inputs. Further details are provided in Appendix B.1.

Table 1: **Performance Comparison.** Results on Qwen2.5-Omni-7B and AudioFlamingo-Next-Instruct. Clotho v2 and AudioCaps report CIDEr×100, while MELD, ClothoAQA, and MMAU report accuracy (%). Rel. Score denotes the average of the benchmark-wise scores normalized by the corresponding Vanilla scores.

Method	Qwen2.5-Omni-7B						AudioFlamingo-Next-Instruct					
	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Rel. Score	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Rel. Score
pruning ratio 0%, retention ratio 100%												
Vanilla	33.3	48.9	58.6	69.5	67.1	100.0	44.6	65.1	26.8	70.8	75.8	100.0
pruning ratio 75%, retention ratio 25%												
VisionZip ('25)	29.3	43.5	58.9	67.8	64.4	94.2	43.6	60.9	28.8	68.5	74.3	98.7
DivPrune ('25)	26.2	39.0	59.5	67.0	63.4	90.2	41.4	54.9	30.3	69.3	72.8	96.8
FLoC ('26)	28.2	41.3	58.9	68.8	66.9	93.7	43.2	59.4	31.2	69.7	74.1	100.2
SpeechPrune ('25)	28.4	36.8	59.1	68.2	64.5	91.1	42.8	50.3	30.4	68.4	73.8	96.1
Seg. Top-K ('26)	29.2	44.0	59.0	67.3	62.7	93.7	43.7	58.1	29.7	68.6	73.9	98.5
Frame ('26)	28.5	43.8	59.1	68.7	65.2	94.4	43.1	57.4	31.3	69.0	73.1	99.2
HeadRouter ('26)	28.7	43.4	58.2	68.0	65.2	93.8	43.5	59.9	28.3	69.1	74.9	98.3
OmniFit ('26)	21.3	41.4	58.7	64.2	58.8	85.8	41.7	59.1	28.1	68.6	69.7	95.6
OmniZip ('26)	29.6	44.1	58.6	67.8	65.1	94.7	43.6	60.5	28.1	69.1	74.6	98.4
TimePrune	29.6	45.8	59.0	68.7	66.9	**96.4**	45.0	62.0	31.4	69.5	75.4	**102.2**
pruning ratio 85%, retention ratio 15%												
VisionZip ('25)	25.3	40.2	58.8	65.8	62.9	89.4	42.2	55.4	29.5	68.0	72.3	96.3
DivPrune ('25)	24.0	34.7	58.9	66.5	61.9	86.3	39.0	49.7	32.0	67.5	71.5	94.6
FLoC ('26)	26.2	37.9	58.8	67.7	62.8	89.5	42.6	56.9	32.3	68.9	71.3	99.0
SpeechPrune ('25)	22.5	25.9	58.9	65.9	61.3	81.5	40.9	37.7	31.5	66.7	71.2	91.1
Seg. Top-K ('26)	23.9	36.5	59.2	66.4	61.1	86.8	42.6	52.5	30.8	68.1	73.4	96.9
Frame ('26)	26.7	37.7	59.6	66.3	61.4	89.2	42.0	56.6	32.3	67.8	71.1	98.3
HeadRouter ('26)	25.1	38.5	58.8	65.2	62.6	88.3	42.1	54.4	30.1	68.0	73.1	96.6
OmniFit ('26)	12.8	36.9	58.7	59.6	56.2	76.7	37.7	53.8	30.3	65.8	68.4	92.7
OmniZip ('26)	25.1	39.1	58.5	66.1	63.6	89.0	41.1	54.8	29.0	67.9	72.6	95.3
TimePrune	27.5	41.8	59.0	67.5	66.0	**92.8**	44.4	56.4	32.8	68.8	75.0	**100.9**
pruning ratio 90%, retention ratio 10%												
VisionZip ('25)	22.1	35.2	58.7	63.6	59.2	83.7	40.7	50.2	30.4	67.3	70.5	94.0
DivPrune ('25)	20.3	29.7	58.7	64.4	57.5	80.1	38.2	45.2	32.6	66.7	70.4	92.8
FLoC ('26)	24.9	34.2	58.7	66.7	62.4	86.8	40.9	55.6	32.9	67.8	69.2	97.4
SpeechPrune ('25)	19.4	17.9	58.9	63.6	57.4	74.5	39.3	27.2	32.0	65.0	68.9	86.4
Seg. Top-K ('26)	21.3	32.4	58.8	63.6	57.5	81.6	40.9	47.5	31.5	67.2	72.3	94.5
Frame ('26)	24.0	34.2	59.2	65.4	59.0	85.0	40.0	46.9	32.3	66.9	70.1	93.9
HeadRouter ('26)	22.3	33.5	58.6	63.6	60.2	83.3	39.9	50.0	31.7	66.9	71.2	94.6
OmniFit ('26)	9.4	35.0	58.7	56.2	54.7	72.5	31.7	43.4	31.8	63.7	67.1	87.0
OmniZip ('26)	22.7	33.8	58.3	64.1	60.0	83.7	39.5	51.5	30.3	66.3	70.9	93.6
TimePrune	24.9	37.9	59.3	66.3	61.8	**88.2**	43.8	52.9	32.9	68.1	71.4	**98.5**

Benchmarks. We evaluate on five benchmarks covering diverse audio understanding tasks: Clotho v2 (Drossos et al., 2020) and AudioCaps (Kim et al., 2019) for audio captioning, MELD (Poria et al., 2019) for speech emotion recognition, ClothoAQA (Lipping et al., 2022) for audio question answering, and MMAU (Sakshi et al., 2025) for audio understanding and reasoning. We report CIDEr scores (Vedantam et al., 2015), multiplied by 100, for Clotho v2 and AudioCaps, and accuracy (%) for the remaining benchmarks. Further details are provided in Appendix B.2.

Compared Methods. We compare TimePrune with representative training-free token compression methods spanning diverse compression principles: VisionZip (Yang et al., 2025), DivPrune (Alvar et al., 2025), FLoC (Cho et al., 2026b), SpeechPrune (Lin et al., 2025), Segmentwise Top-K (Gibier et al., 2026), Frame (He et al., 2025), HeadRouter (He et al., 2026), OmniFit (Wang et al., 2026), and OmniZip (Tao et al., 2026). For methods originally designed for multimodal compression, such as OmniFit and OmniZip, we apply their compression mechanisms only to the audio modality to maintain a consistent audio token pruning setting. The selected methods cover complementary criteria including saliency, redundancy, diversity, temporal structure, and task-adaptive relevance, with implementation details provided in Appendix B.3.

4.2 Main Results

Main Performance Comparison. Table 1 compares TimePrune with existing token compression methods on Qwen2.5-Omni-7B (Xu et al., 2025a) and AudioFlamingo-Next-Instruct (Ghosh et al., 2026), both using Whisper-based audio encoders despite different architectures and training strategies. Across diverse audio understanding tasks and pruning ratios, TimePrune maintains competitive downstream performance under constrained token budgets.

Results across Different Audio Encoder Architectures. Table 2 extends the comparison to Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025b), which uses an Audio Transformer (AuT) encoder, and MOSS-Audio-8B-Instruct (Yang et al., 2026), which uses a sliding-window Transformer audio encoder. Without additional training or model-specific adaptation, TimePrune maintains com-

Table 2: **Performance Comparison.** Results on Qwen3-Omni-30B-A3B-Instruct and MOSS-Audio-8B-Instruct. MELD reports accuracy (%), while ClothoAQA reports Unanswerable (Unan.) and Non-binary (Non-bin.) accuracy, and MMAU reports accuracy for Sound, Speech, and Music. Rel. Score denotes the average of the benchmark-wise scores normalized by the corresponding Vanilla scores.

Method	Qwen3-Omni-30B-A3B-Instruct							MOSS-Audio-8B-Instruct						
	MELD	ClothoAQA		MMAU			Rel. Score	MELD	ClothoAQA		MMAU			Rel. Score
		Unan.	*Non-bin.*	*Sound*	*Speech*	*Music*			*Unan.*	*Non-bin.*	*Sound*	*Speech*	*Music*	
pruning ratio 0%, retention ratio 100%														
Vanilla	51.4	70.8	40.9	80.2	78.4	70.7	100.0	57.8	84.9	33.7	78.7	67.0	78.4	100.0
pruning ratio 85%, retention ratio 15%														
VisionZip ('25)	57.5	67.2	37.4	76.3	56.8	70.4	98.1	60.5	81.0	30.5	75.4	61.3	75.7	97.8
DivPrune ('25)	56.7	65.9	37.3	76.6	61.0	70.1	97.7	61.3	81.7	30.5	73.0	60.1	75.7	97.9
FLoC ('26)	56.8	68.0	40.0	75.7	60.6	70.4	99.1	60.8	82.1	31.5	73.6	61.0	75.4	98.2
SpeechPrune ('25)	57.1	67.0	39.5	74.2	57.6	69.8	98.1	60.7	80.5	29.9	73.3	56.5	77.2	96.8
Seg. Top-K ('26)	58.1	67.5	36.4	75.1	52.7	71.3	97.6	61.1	80.8	31.4	73.6	60.7	76.0	98.1
Frame ('26)	57.7	67.5	37.8	77.8	51.5	71.0	98.0	61.0	82.6	31.9	76.3	56.8	76.0	98.5
HeadRouter ('26)	61.1	73.3	42.2	75.1	58.7	72.8	104.1	60.9	79.7	31.1	74.8	61.0	77.5	98.0
OmniFit ('26)	53.3	67.4	38.4	81.4	53.0	71.9	96.1	60.5	82.1	31.2	73.3	55.0	75.7	97.0
OmniZip ('26)	58.2	66.5	37.3	74.8	55.3	69.8	97.8	60.9	81.3	31.7	75.4	61.3	76.6	98.6
TimePrune	60.8	72.2	41.8	80.8	58.0	72.4	**104.2**	61.6	82.7	32.1	78.4	62.7	79.7	**100.6**
pruning ratio 90%, retention ratio 10%														
VisionZip ('25)	58.2	67.5	36.8	72.7	51.1	71.3	97.2	60.9	79.1	29.9	74.8	60.4	76.3	97.2
DivPrune ('25)	57.1	66.8	36.0	75.4	53.0	70.7	96.7	60.5	80.4	30.1	73.9	58.0	77.2	97.1
FLoC ('26)	57.9	69.4	39.5	75.7	54.2	71.0	99.2	60.7	81.2	30.8	74.2	56.5	76.9	97.4
SpeechPrune ('25)	57.6	67.0	37.4	73.0	49.6	69.5	96.4	61.0	80.1	29.1	71.5	54.7	76.3	96.0
Seg. Top-K ('26)	58.4	69.0	35.2	75.7	52.3	70.7	97.8	60.6	80.1	29.9	73.3	57.1	75.7	96.5
Frame ('26)	58.1	67.9	36.6	74.5	48.1	71.0	97.0	61.4	80.7	30.8	74.5	52.6	75.1	96.8
HeadRouter ('26)	61.1	70.7	39.7	73.6	54.2	73.7	101.9	60.5	79.0	30.3	74.2	57.7	76.3	96.6
OmniFit ('26)	54.4	65.9	37.6	79.0	48.5	69.5	94.8	60.7	79.9	28.9	71.5	54.1	74.9	95.4
OmniZip ('26)	58.2	68.8	37.7	74.5	54.9	70.1	98.5	61.6	80.0	30.3	75.1	58.6	76.9	97.9
TimePrune	60.5	70.6	39.6	78.4	53.8	73.0	**102.0**	61.5	81.9	31.4	74.5	57.1	76.9	**98.4**

petitive performance across pruning ratios, demonstrating its applicability across distinct audio representation pipelines. Results under more aggressive pruning ratios are provided in Appendix A.1.

4.3 Ablation Studies

Effect of Temporal Coverage Refinement. We vary β in Table 3 while fixing the time-synchronized representation deviation criterion and cosine distance, where $\beta = \infty$ corresponds to Time-Aware RDP without Temporal Coverage Refinement. Avg. TGR denotes the mean TGR across the three benchmarks, while Avg. Score denotes the mean of the reported benchmark scores. Introducing coverage refinement substantially reduces Avg. TGR from 5.584 at $\beta = \infty$ to 2.676 at $\beta = 2$, while increasing Avg. Score from 46.4 to 48.7. Further decreasing β to 1 provides negligible additional reduction in Avg. TGR while slightly decreasing Avg. Score. Additional results across different ALLMs are provided in Appendix A.2.

Effect of Trajectory-Driven Refinement. To examine the role of trajectory information in anchor selection, Table 4 compares Time-Sync. Deviation with two alternatives under the same refinement setting. Midpoint selects the center of each interval, while Adjacent Change selects the token with the largest local representation change. Time-Sync. Deviation instead selects the representation least sufficiently described by the current anchors. It achieves the highest Avg. Score of 48.7, compared with 46.7 and 47.2 for Midpoint and Adjacent Change, respectively. These results support trajectory deviation as the refinement criterion in TimePrune. Additional results across different ALLMs and details of the alternative refinement criteria are provided in Appendix A.3.

Table 3: **Effect of Temporal Coverage Refinement.** Results at 90% pruning with varying β.

β	Avg. TGR ↓	Clotho v2	MELD	MMAU	Avg. Score ↑
∞	5.584	20.1	59.0	60.1	46.4
5.0	4.128	21.4	59.1	60.8	47.1
4.0	3.497	22.9	59.0	61.6	47.8
3.0	2.834	22.9	59.3	59.8	47.3
2.0	2.676	**24.9**	59.3	**61.8**	**48.7**
1.0	**2.675**	24.1	**59.5**	60.6	48.1

Table 4: **Effect of the Refinement Criterion.** Results on three representative benchmarks at 90% pruning with $\beta = 2$. Avg. Score denotes the mean of the reported benchmark scores.

Refinement	Clotho v2	MELD	MMAU	Avg. Score
Midpoint	22.1	**59.3**	58.8	46.7
Adjacent Change	21.4	59.2	60.9	47.2
Time-Sync. Deviation	**24.9**	**59.3**	**61.8**	**48.7**

Representation Deviation Metric. We compare ℓ_1 distance, ℓ_2 distance, cosine distance, and negative dot product while fixing Temporal Coverage Refinement at $\beta = 2$. As shown in Table 5, the choice of deviation metric leads to relatively modest performance differences across the three benchmarks, with cosine distance achieving the highest Avg. Score of 48.7. We therefore use cosine distance for Qwen2.5-Omni-7B. Detailed metric definitions and additional results across different ALLMs are provided in Appendix A.4.

Table 5: **Effect of the Representation Deviation Metric.** Results on three representative benchmarks at 90% pruning with $\beta = 2$. Avg. Score denotes the mean of the reported benchmark scores.

Deviation Metric	Clotho v2	MELD	MMAU	Avg. Score
ℓ_1 Distance	24.1	58.7	61.6	48.1
ℓ_2 Distance	22.6	**59.4**	60.7	47.6
Cosine Distance	**24.9**	59.3	**61.8**	**48.7**
Negative Dot Product	22.9	59.0	59.9	47.3

4.4 Efficiency Analysis

We further examine whether the computational savings from audio token reduction translate into practical inference efficiency when accounting for the cost of token compression. We evaluate TimePrune on Qwen2.5-Omni-7B using MMAU test-mini (1,000 samples), including compression overhead in the total prefilling time. As shown in Table 6, TimePrune retains 98.4% of the Vanilla score at 15% token retention, exceeding Segmentwise Top-K (97.2%) at 30% retention. We use 15% retention for TimePrune to evaluate whether it can preserve comparable performance under a more aggressive token budget than the baselines. This reduces audio-token-dependent LLM floating-point operations (FLOPs) from 2.36T to 0.35T and the key–value (KV) cache from 19.6 MB to 2.9 MB, while achieving a $1.98\times$ total prefilling speedup despite the compression overhead. Additional efficiency results across different ALLMs are provided in Appendix F.

Table 6: **Inference efficiency on Qwen2.5-Omni-7B using MMAU test-mini.** FLOPs denote the theoretical audio-token-dependent LLM computation. Total includes compression overhead, and Rel. Score is measured relative to Vanilla.

Method	Retention Ratio R	FLOPs (T)	Prefilling Time			KV Cache (MB)	Score	Rel. Score (%)
			Compression	LLM Forward	Total			
Vanilla	100%	2.36	–	48.2	48.2 (1.0×)	19.6	0.671	100.0
VisionZip	30%	0.71	4.2	26.1	30.3 (1.59×)	5.9	0.649	96.7
DivPrune	30%	0.70	3.6	25.9	29.5 (1.63×)	5.9	0.648	96.6
Seg. Top-K	30%	0.67	**1.8**	25.7	27.5 (1.75×)	5.6	0.652	97.2
HeadRouter	30%	0.70	**1.8**	25.9	27.7 (1.74×)	5.9	0.650	96.9
TimePrune	**15%**	**0.35**	2.4	**21.9**	**24.3 (1.98×)**	**2.9**	0.660	**98.4**

5 Conclusion

We presented TimePrune, a training-free audio token pruning framework that preserves temporal representation trajectories through adaptive temporal resolution. TimePrune combines Time-Aware RDP, which recursively retains temporal anchors based on time-synchronized representation deviation, with Temporal Coverage Refinement, which prioritizes the refinement of excessively sparse temporal intervals. Experiments across multiple ALLMs and diverse audio understanding benchmarks demonstrate that TimePrune maintains competitive downstream performance under aggressive token compression while substantially reducing inference cost. Our analyses further highlight the complementary roles of trajectory-driven refinement and temporal coverage in preserving temporal representation structure under limited token budgets, providing a trajectory-preservation perspective for efficient audio token compression in ALLMs.

AI Use Statement

Generative AI tools were used to assist with research ideation and execution, literature retrieval and discovery, drafting parts of the manuscript, and improving the clarity and readability of the

writing. The authors critically reviewed and verified all AI-assisted outputs, including research-related suggestions, retrieved information, and generated or edited text. All research decisions, experimental results, analyses, and conclusions were independently validated by the authors. The authors take full responsibility for the final content of this work.

Reproducibility statement

We provide detailed experimental settings, including model-specific implementation details, dataset and evaluation protocols, and baseline implementations, in Appendix B. Additional ablation studies are provided in Appendix A, and further efficiency analyses are provided in Appendix F. These details are intended to facilitate reproduction of the experimental results reported in this work.

References

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9392–9401, 2025. doi: 10.1109/CVPR52734.2025.00877. URL https://openaccess.thecvf.com/content/CVPR2025/html/Alvar_DivPrune_Diversity-based_Visual_Token_Pruning_for_Large_Multimodal_Models_CVPR_2025_paper.html.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. doi: 10.48550/arXiv.2502.13923. URL https://arxiv.org/abs/2502.13923.

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 *Computer Vision – ECCV 2024*, pp. 19–35. Springer, 2024. doi: 10.1007/978-3-031-73004-7_2. URL https://doi.org/10.1007/978-3-031-73004-7_2.

Hyeonwoo Cho, Donghyeon Baek, Yewon Kim, and Bumsub Ham. Improving visual token reduction via rectifying distortions for efficient multimodal llm inference. In *International Conference on Machine Learning*, 2026a.

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 *The Fourteenth International Conference on Learning Representations*, 2026b.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*, 2023. doi: 10.48550/arXiv.2311.07919. URL https://arxiv.org/abs/2311.07919.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*, 2024. doi: 10.48550/arXiv.2407.10759. URL https://arxiv.org/abs/2407.10759.

Junbo Cui, Bokai Xu, Chongyi Wang, Tianyu Yu, Weiyue Sun, Yingjing Xu, Tianran Wang, Zhihui He, Wenshuo Ma, Tianchi Cai, Jiancheng Gui, Luoyuan Zhang, Xian Sun, Fuwei Huang, Moye Chen, Zhuo Lin, Hanyu Liu, Qingxin Gui, Qingzhe Han, Yuyang Wen, Huiping Liu, Rongkang Wang, Yaqi Zhang, Hongliang Wei, Chi Chen, You Li, Kechen Fang, Jie Zhou, Yuxuan Li, Guoyang Zeng, Chaojun Xiao, Yankai Lin, Xu Han, Maosong Sun, Zhiyuan Liu, and Yuan Yao. Minicpm-o 4.5: Towards real-time full-duplex omni-modal interaction, 2026.

Jinhong Deng, Wen Li, Joey Tianyi Zhou, and Yang He. Scope: Saliency-coverage oriented token pruning for efficient multimodel llms. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5394.

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 *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

Sixun Dong, Juhua Hu, Mian Zhang, Ming Yin, Yanjie Fu, and Qi Qian. Mmtok: Multimodal coverage maximization for efficient inference of vlms. In *The Fourteenth International Conference on Learning Representations*, 2026.

David H. Douglas and Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, 1973. doi: 10.3138/FM57-6770-U75U-7727. URL https://doi.org/10.3138/FM57-6770-U75U-7727.

Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. Clotho: An audio captioning dataset. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 736–740. IEEE, 2020. doi: 10.1109/ICASSP40776.2020.9052990. URL https://doi.org/10.1109/ICASSP40776.2020.9052990.

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 *The Fourteenth International Conference on Learning Representations*, 2026.

Sreyan Ghosh, Zhifeng Kong, Sonal Kumar, S Sakshi, Jaehyeon Kim, Wei Ping, Rafael Valle, Dinesh Manocha, and Bryan Catanzaro. Audio flamingo 2: An audio-language model with long-audio understanding and expert reasoning abilities. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19358–19405. PMLR, 2025. URL https://proceedings.mlr.press/v267/ghosh25b.html.

Sreyan Ghosh, Arushi Goel, Kaousheik Jayakumar, Lasha Koroshinadze, Nishit Anand, Zhifeng Kong, Siddharth Gururani, Sang gil Lee, Jaehyeon Kim, Aya Aljafari, Chao-Han Huck Yang, Sungwon Kim, Ramani Duraiswami, Dinesh Manocha, Mohammad Shoeybi, Bryan Catanzaro, Ming-Yu Liu, and Wei Ping. Audio flamingo next: Next-generation open audio-language models for speech, sound, and music, 2026.

Marcel Gibier, Raphaël Duroselle, Pierre Serrano, Olivier Boeffard, and Jean-François Bonastre. Segmentwise pruning in audio-language models. In *ICASSP 2026 – 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2026. doi: 10.1109/ICASSP55912.2026.11463308. URL https://doi.org/10.1109/ICASSP55912.2026.11463308.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=FjByDpDVIO.

Peize He, Zichen Wen, Yubo Wang, Yuxuan Wang, Xiaoqian Liu, Jiajie Huang, Zehui Lei, Zhuangcheng Gu, Xiangqi Jin, Jiabing Yang, Kai Li, Zhifei Liu, Weijia Li, Cunxiang Wang, Conghui He, and Linfeng Zhang. Audiomarathon: A comprehensive benchmark for long-context audio understanding and efficiency in audio llms. *arXiv preprint arXiv:2510.07293*, 2025. URL https://arxiv.org/abs/2510.07293.

Peize He, Yaodi Luo, Xiaoqian Liu, Xuyang Liu, Jiahang Deng, Yaosong Du, Bangyu Li, Xiyan Gui, Yuxuan Chen, and Linfeng Zhang. Headrouter: Dynamic head-weight routing for task-adaptive audio token pruning in large audio language models. In *Proceedings of the 34th ACM International Conference on Multimedia*, 2026.

Inclusion AI. Ming-omni: A unified multimodal model for perception and generation, 2025.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 119–132. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1011. URL https://aclanthology.org/N19-1011/.

Youngeun Kim, Youjia Zhang, Huiling Liu, Aecheon Jung, Sunwoo Lee, and Sungeun Hong. Zoo-prune: Training-free token pruning via zeroth-order gradient estimation in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39572–39582, 2026. doi: 10.1109/CVPR55763.2026.03860. URL https://openaccess.thecvf.com/content/CVPR2026/html/Kim_ZOO-Prune_Training-Free_Token_Pruning_via_Zeroth-Order_Gradient_Estimation_in_Vision-Language_CVPR_2026_paper.html.

KimiTeam, Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, et al. Kimi-audio technical report, 2025.

Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 25125–25148. PMLR, 2024. URL https://proceedings.mlr.press/v235/kong24a.html.

Sangin Lee and Yukyung Choi. Clip tricks you: Training-free token pruning for efficient pixel grounding in large vision-language models. In *International Conference on Machine Learning*, 2026.

Ao Li, Yuxiang Duan, Jinghui Zhang, Congbo Ma, Yutong Xie, Gustavo Carneiro, Mohammad Yaqub, and Hu Wang. Transprune: Token transition pruning for efficient large vision-language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39529–39538, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Li_TransPrune_Token_Transition_Pruning_for_Efficient_Large_Vision-Language_Model_CVPR_2026_paper.html.

Bingzhou Li and Tao Huang. Dash: Dynamic audio-driven semantic chunking for efficient omnimodal token compression. *arXiv preprint arXiv:2603.15685*, 2026. doi: 10.48550/arXiv.2603.15685. URL https://arxiv.org/abs/2603.15685.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a. doi: 10.48550/arXiv.2408.03326. URL https://arxiv.org/abs/2408.03326.

Yadong Li, Haoze Sun, Mingan Lin, Tianpeng Li, Guosheng Dong, Tao Zhang, Bowen Ding, Wei Song, Zhenglin Cheng, Yuqi Huo, et al. Baichuan-omni technical report, 2024b.

Xuelian Lin, Jiahao Jiang, Shuai Ma, Yimeng Zuo, and Chunming Hu. One-pass trajectory simplification using the synchronous euclidean distance. *The VLDB Journal*, 28(6):897–921, 2019. doi: 10.1007/s00778-019-00575-8.

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)*. IEEE, 2025. doi: 10.1109/ICME59968.2025.11209113. URL https://doi.org/10.1109/ICME59968.2025.11209113.

Samuel Lipping, Parthasaarathy Sudarsanam, Konstantinos Drossos, and Tuomas Virtanen. Clotho-aqa: A crowdsourced dataset for audio question answering. *arXiv preprint arXiv:2204.09634*, 2022. doi: 10.48550/arXiv.2204.09634. URL https://arxiv.org/abs/2204.09634.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023.

Jiale Luo, Xiaoyu Liang, and Haoji Hu. Locality matters for training-free audio token compression in audio-language models, 2026.

Nirvana Meratnia and Rolf A. de By. Spatiotemporal compression techniques for moving point objects. In *Advances in Database Technology – EDBT 2004*, volume 2992 of *Lecture Notes in Computer Science*, pp. 765–782. Springer, 2004. doi: 10.1007/978-3-540-24741-8_44. URL https://doi.org/10.1007/978-3-540-24741-8_44.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. Meld: A multimodal multi-party dataset for emotion recognition in conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 527–536. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1050. URL https://aclanthology.org/P19-1050/.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 28492–28518. PMLR, 2023.

Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244–256, 1972. doi: 10.1016/S0146-664X(72)80017-0. URL https://doi.org/10.1016/S0146-664X(72)80017-0.

S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. Mmau: A massive multi-task audio understanding and reasoning benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TeVAZXr3yv.

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.

Wanshun Su, Yang Shi, Feihu Liu, Ziwen Yu, Yan Min, Zhuoran Zhang, Qixun Wang, Haotian Wang, Shixuan Liu, Yuanxing Zhang, Peng Wu, Chengfu Huo, and Liang Ding. Omnipack: Unified token compression for efficient omni-modal large language models, 2026.

Rajesh Tamilmani and Emmanuel Stefanakis. Semantically enriched simplification of trajectories. *Proceedings of the ICA*, 2:128, 2019. doi: 10.5194/ica-proc-2-128-2019. URL https://doi.org/10.5194/ica-proc-2-128-2019.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. Salmonn: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=14rn7HpKVk.

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 *The Fourteenth International Conference on Learning Representations*, 2026.

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 (CVPR)*, pp. 17682–17692, 2026. doi: 10.1109/CVPR55763.2026.01725. URL https://openaccess.thecvf.com/content/CVPR2026/html/Tao_OmniZip_Audio-Guided_Dynamic_Token_Compression_for_Fast_Omnimodal_Large_Language_CVPR_2026_paper.html.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4566–4575, 2015.

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 *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

Zichen Wen, Yifeng Gao, Shaobo Wang, Junyuan Zhang, Qintong Zhang, Weijia Li, Conghui He, and Linfeng Zhang. Stop looking for "important tokens" in multimodal language models: Duplication matters more. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9961–9980. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.505. URL https://aclanthology.org/2025.emnlp-main.505/.

Xiaomi LLM-Core Team, Dong Zhang, Gang Wang, Jinlong Xue, Kai Fang, Liang Zhao, Rui Ma, Shuhuai Ren, Shuo Liu, Tao Guo, et al. Mimo-audio: Audio language models are few-shot learners, 2025.

Zijie Xin, Jie Yang, Ruixiang Zhao, Tianyi Wang, Fengyun Rao, Jing Lyu, and Xirong Li. Stage-adaptive token selection for efficient omni-modal llms. *arXiv preprint arXiv:2605.20035*, 2026.

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, 2025a.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-omni technical report, 2025b.

Tong Xu, Hailong Shi, and Xingyu Gao. Score: Salience-coverage reduction for vision token pruning in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24686–24695, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Xu_SCoRe_Salience-Coverage_Reduction_for_Vision_Token_Pruning_in_Vision-Language_Models_CVPR_2026_paper.html.

Chen Yang, Chufan Yu, Hanfu Chen, Jie Zhu, Jingqi Chen, Ke Chen, Wenxuan Wang, Yang Wang, Yaozhou Jiang, Yi Jiang, Zhengyuan Lin, Ziqi Chen, Zhaoye Fei, Chenghao Liu, Donghua Yu, Jun Zhan, Kang Yu, Kexin Huang, Liwei Fan, Mingshu Chen, Qinyuan Cheng, Ruixiao Li, Shimin Li, Songlin Wang, Xingjian Zhao, Yang Gao, Yitian Gong, Yiyang Zhang, Zhe Xu, and Xipeng Qiu. Moss-audio technical report, 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802, 2025. doi: 10.1109/CVPR52734.2025.01843. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yang_VisionZip_Longer_is_Better_but_Not_Necessary_in_Vision_Language_CVPR_2025_paper.html.

Qizhe Zhang, Aosong Cheng, Ming Lu, Zhiyong Zhuo, Minqi Wang, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. [cls] attention is all you need for training-free visual token pruning: Make vlm inference faster. *arXiv preprint arXiv:2412.01818*, 2024. doi: 10.48550/arXiv.2412.01818. URL https://arxiv.org/abs/2412.01818.

Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond text-visual attention: Exploiting visual cues for effective token pruning in vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20857–20867, 2025a. doi: 10.1109/ICCV51701.2025.01939. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhang_Beyond_Text-Visual_Attention_Exploiting_Visual_Cues_for_Effective_Token_Pruning_ICCV_2025_paper.html.

Qizhe Zhang, Mengzhen Liu, Lichen Li, Ming Lu, Yuan Zhang, Junwen Pan, Qi She, and Shanghang Zhang. Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. In *Advances in Neural Information Processing Systems*, volume 38, 2025b. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/2433fec2144ccf5fea1c9c5ebdbc3924-Abstract-Conference.html.

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 (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 68588–68607. PMLR, 2025c. URL https://proceedings.mlr.press/v267/zhang25s.html.

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.

Supplementary Material

CONTENTS

A Additional Experimental Results **17**

A.1 Results under Additional Pruning Ratios 17

A.2 Cross-Model Analysis of Temporal Coverage Refinement 18

A.3 Cross-Model Analysis of Trajectory-Driven Refinement 19

A.4 Cross-Model Analysis of Representation Deviation Metrics 20

B Detailed Experimental Setup **21**

B.1 Models 21

B.2 Datasets and Evaluation Protocols 22

B.3 Compared Methods 23

C Additional Related Work **24**

C.1 Token Pruning and Compression 24

C.2 Trajectory Simplification and RDP 25

D Additional Analysis of Temporal Trajectory Preservation **26**

D.1 Effect of Temporal Coverage Refinement across Pruning Ratios 26

D.2 Cross-Method Analysis of Trajectory Fidelity and Temporal Coverage 27

D.3 Visualization of Temporal Trajectory Preservation 27

D.4 Additional Temporal Coverage Analysis 28

E Additional Analysis of Adaptive Temporal Resolution **29**

E.1 Cross-Method Analysis of Adaptive Temporal Allocation 29

E.2 Effect of Temporal Coverage Refinement on Token Allocation 30

E.3 Sample-Level Visualization of Adaptive Temporal Allocation 31

F Additional Efficiency Analysis **31**

F.1 Efficiency Measurement Details 31

F.2 Efficiency across Different ALLMs 33

A Additional Experimental Results

In this section, we provide additional experiments that complement the main results across token budgets, benchmarks, and model architectures. In Sec. A.1, we extend the evaluation of Qwen2.5-Omni, AudioFlamingo-Next-Instruct, Qwen3-Omni, and MOSS-Audio to more aggressive pruning settings, examining the behavior of TimePrune when only a small fraction of the original audio tokens is retained. The remaining subsections extend the ablation analyses presented for Qwen2.5-Omni-7B in the main paper to a broader set of benchmarks and additional ALLMs. Specifically, Sec. A.2 examines Temporal Coverage Refinement, Sec. A.3 analyzes trajectory-driven refinement, and Sec. A.4 evaluates different representation deviation metrics across different model architectures and audio representation pipelines. Together, these experiments provide a broader evaluation of TimePrune under aggressive token compression and across different benchmarks and ALLMs.

A.1 Results under Additional Pruning Ratios

Results on Qwen2.5-Omni-7B and AudioFlamingo-Next-Instruct. As shown in Table 7, TimePrune maintains strong relative performance as the token budget is further reduced. On Qwen2.5-Omni-7B, TimePrune achieves Rel. Scores of 85.2% and 83.5% at 92.5% and 95% pruning, respectively, achieving the highest Rel. Score among the compared methods at both pruning ratios. On AudioFlamingo-Next-Instruct, TimePrune achieves Rel. Scores of 96.1% and 92.2%, respectively, with the highest Rel. Score at 92.5% pruning and performance comparable to the best-performing baseline at 95% pruning. These results demonstrate that TimePrune remains effective under highly constrained token budgets, even when only 7.5% or 5% of the original audio tokens are retained.

Table 7: **Performance Comparison under Aggressive Pruning.** Results on Qwen2.5-Omni-7B and AudioFlamingo-Next-Instruct at 92.5% and 95% pruning ratios. Clotho v2 and AudioCaps report CIDEr$\times$100, while MELD, ClothoAQA, and MMAU report accuracy (%). Rel. Score denotes the average of the benchmark-wise scores normalized by the corresponding Vanilla scores.

Method	Qwen2.5-Omni-7B						AudioFlamingo-Next-Instruct					
	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Rel. Score	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Rel. Score
pruning ratio 0%, retention ratio 100%												
Vanilla	34.0	48.9	58.6	69.5	67.5	100.0	44.7	65.9	26.8	70.7	75.7	100.0
pruning ratio 92.5%, retention ratio 7.5%												
VisionZip ('25)	19.1	30.7	58.5	63.2	57.5	79.0	36.6	45.9	31.0	65.4	69.3	90.2
DivPrune ('25)	17.6	26.3	58.6	62.4	60.1	76.9	34.3	42.0	32.7	63.6	69.9	88.9
FLoC ('26)	23.5	31.2	58.8	65.0	60.5	83.3	38.6	52.2	33.1	66.6	69.1	94.9
SpeechPrune ('25)	16.9	13.7	58.8	61.9	56.4	70.2	35.3	21.4	31.6	62.7	68.4	81.7
Seg. Top-K ('26)	17.6	24.8	58.9	62.1	56.0	75.1	37.2	39.5	32.2	65.7	70.8	90.0
Frame ('26)	21.3	30.4	58.6	64.9	57.8	80.8	37.9	47.2	32.3	66.3	69.6	92.5
HeadRouter ('26)	18.2	28.6	58.7	61.6	58.0	77.3	35.6	45.4	32.4	64.6	69.0	90.4
OmniFit ('26)	16.8	31.8	58.6	61.9	56.4	77.5	36.7	37.4	32.0	65.2	67.5	87.9
OmniZip ('26)	19.2	27.8	58.4	63.2	58.2	78.0	35.9	44.4	31.2	65.1	68.8	89.4
TimePrune	22.9	37.1	58.9	65.3	59.5	**85.2**	40.4	51.8	33.0	66.9	70.9	**96.1**
pruning ratio 95%, retention ratio 5%												
VisionZip ('25)	12.8	21.4	59.3	59.1	55.4	69.9	29.3	36.3	31.2	61.8	68.0	82.9
DivPrune ('25)	15.1	22.8	58.7	60.6	56.1	72.3	32.0	32.8	32.7	60.7	69.0	84.1
FLoC ('26)	21.5	29.8	58.6	63.7	58.2	80.4	35.2	50.2	33.5	64.2	68.9	**92.4**
SpeechPrune ('25)	13.7	9.7	58.7	58.9	53.5	64.8	28.9	14.2	32.4	59.5	67.3	76.0
Seg. Top-K ('26)	10.0	24.3	59.1	55.9	54.5	68.2	32.9	39.3	32.3	62.7	69.0	86.7
Frame ('26)	18.7	25.2	58.9	62.9	55.9	76.1	35.8	42.3	32.0	64.3	68.0	88.8
HeadRouter ('26)	11.2	20.4	59.1	57.2	55.1	67.9	28.7	36.3	32.6	61.1	68.2	83.4
OmniFit ('26)	16.0	28.9	58.7	62.0	53.4	75.0	33.1	30.3	32.7	63.6	66.5	84.0
OmniZip ('26)	17.6	22.1	59.3	61.9	55.5	73.8	32.3	33.2	32.0	62.0	68.1	83.9
TimePrune	22.7	35.2	59.3	64.7	57.2	**83.5**	38.2	46.0	32.8	64.9	69.5	92.2

Results on Qwen3-Omni and MOSS-Audio. Table 8 extends the evaluation to Qwen3-Omni-30B-A3B-Instruct and MOSS-Audio-8B-Instruct under the same aggressive pruning settings. On Qwen3-Omni, TimePrune achieves Rel. Scores of 100.8% and 99.5% at 92.5% and 95% pruning, respectively, achieving the highest Rel. Score among the compared methods at both pruning ratios. On MOSS-Audio, TimePrune achieves Rel. Scores of 97.4% and 96.1%, respectively, while retaining only 7.5% and 5% of the original audio tokens. It achieves the highest Rel. Score among the compared methods at both pruning ratios. These results further demonstrate the effectiveness of TimePrune under highly constrained token budgets across different ALLMs.

Table 8: **Performance Comparison under Aggressive Pruning.** Results on Qwen3-Omni-30B-A3B-Instruct and MOSS-Audio-8B-Instruct at 92.5% and 95% pruning ratios. MELD reports accuracy (%), while ClothoAQA reports Unanswerable (Unan.) and Non-binary (Non-bin.) accuracy, and MMAU reports accuracy for Sound, Speech, and Music. Rel. Score denotes the average of the benchmark-wise scores normalized by the corresponding Vanilla scores.

Method	Qwen3-Omni-30B-A3B-Instruct							MOSS-Audio-8B-Instruct						
	MELD	ClothoAQA		MMAU			Rel. Score	MELD	ClothoAQA		MMAU			Rel. Score
		Unan.	*Non-bin.*	*Sound*	*Speech*	*Music*			*Unan.*	*Non-bin.*	*Sound*	*Speech*	*Music*	
pruning ratio 0%, retention ratio 100%														
Vanilla	51.4	70.8	40.9	80.2	78.4	70.7	100.0	57.8	84.9	33.7	78.7	67.0	78.4	100.0
pruning ratio 92.5%, retention ratio 7.5%														
VisionZip ('25)	58.5	68.4	35.8	73.9	51.5	70.4	97.5	60.3	78.9	29.9	74.2	58.0	74.9	96.2
DivPrune ('25)	57.7	67.1	36.2	73.3	49.6	68.9	96.1	60.5	80.0	29.2	71.8	55.6	74.0	95.5
FLoC ('26)	57.9	68.0	37.5	75.4	50.0	70.7	97.5	60.2	80.4	30.1	72.4	53.5	74.9	95.6
SpeechPrune ('25)	58.4	67.5	37.0	73.6	50.4	69.8	97.2	60.5	78.9	27.9	69.7	54.4	74.3	94.4
Seg. Top-K ('26)	58.9	67.5	33.9	73.3	48.9	69.8	96.4	59.9	78.0	29.7	73.3	56.2	74.9	95.2
Frame ('26)	58.7	66.3	35.9	75.4	47.0	70.7	96.6	61.2	79.7	30.4	74.8	52.0	73.7	96.1
HeadRouter ('26)	61.1	70.2	39.0	72.4	51.9	71.6	100.7	59.9	77.8	29.1	73.6	56.8	74.9	95.1
OmniFit ('26)	54.6	65.5	36.9	77.2	49.2	71.0	94.7	60.2	78.5	28.4	70.3	55.0	72.2	94.1
OmniZip ('26)	58.9	69.1	36.6	74.2	50.4	70.1	98.0	61.3	78.7	29.3	73.9	56.2	75.1	96.2
TimePrune	60.4	70.1	38.9	77.0	51.8	71.4	**100.8**	61.4	81.1	30.8	73.9	56.5	75.1	**97.4**
pruning ratio 95%, retention ratio 5%														
VisionZip ('25)	58.8	65.6	33.3	71.5	48.5	70.4	95.3	60.3	77.3	27.6	71.8	55.0	74.3	94.2
DivPrune ('25)	57.0	66.2	34.6	72.1	48.9	69.8	94.8	59.8	78.4	27.8	70.6	55.0	74.9	94.2
FLoC ('26)	58.4	67.6	37.0	73.9	48.9	69.5	97.0	59.6	79.5	28.9	71.2	52.6	73.7	94.2
SpeechPrune ('25)	58.1	66.7	35.5	73.3	48.1	70.4	96.1	59.8	74.9	26.4	70.3	50.8	74.6	92.1
Seg. Top-K ('26)	58.8	65.9	32.2	70.0	51.1	67.1	94.8	58.6	75.9	27.4	72.7	53.8	73.7	92.6
Frame ('26)	59.2	65.5	35.2	75.4	46.2	70.4	96.4	61.2	77.7	28.9	73.6	49.8	73.1	94.5
HeadRouter ('26)	60.9	69.1	38.7	71.5	50.4	68.9	99.4	58.6	75.6	26.6	72.1	55.3	74.9	92.6
OmniFit ('26)	55.4	65.2	37.8	79.0	48.5	70.7	95.5	59.8	77.1	26.6	72.7	53.5	74.3	93.5
OmniZip ('26)	59.3	66.2	32.1	73.6	48.1	71.3	95.8	61.3	78.6	29.8	73.0	53.5	73.4	95.6
TimePrune	60.2	69.0	38.6	75.2	49.8	70.0	**99.5**	61.1	79.0	30.5	72.4	55.6	74.3	**96.1**

A.2 Cross-Model Analysis of Temporal Coverage Refinement

We extend the analysis of Temporal Coverage Refinement presented in the main paper by providing results across a broader set of benchmarks and additional ALLMs with different audio representation pipelines. We vary $\beta \in \{\infty, 5.0, 4.0, 3.0, 2.0, 1.0\}$ while keeping the time-synchronized representation deviation criterion and ℓ_2 distance fixed, where $\beta = \infty$ corresponds to Time-Aware RDP without Temporal Coverage Refinement. For Qwen2.5-Omni-7B, we additionally report results on ClothoAQA and AudioCaps beyond the representative benchmarks considered in the main ablation, and use the same benchmark set for the cross-model analysis. This controlled comparison examines the effect of Temporal Coverage Refinement on temporal coverage and downstream performance across different audio representation spaces.

Qwen2.5-Omni-7B. Table 9 extends the main analysis of Temporal Coverage Refinement to five benchmarks. Increasing the strength of coverage refinement progressively reduces Avg. TGR from 5.584 at $\beta = \infty$ to 2.676 at $\beta = 2$, while Avg. Score increases from 46.4 to 50.0. Further decreasing β to 1 produces nearly identical Avg. TGR with a slightly lower Avg. Score of 48.3, indicating stable refinement behavior around $\beta = 2$. Overall, Temporal Coverage Refinement effectively regulates excessive temporal sparsity while maintaining downstream performance across the evaluated benchmarks.

Table 9: **Effect of Temporal Coverage Refinement on Qwen2.5-Omni-7B.** Results across five benchmarks at 90% pruning with Time-Sync. Deviation and cosine distance fixed. Avg. TGR denotes the mean TGR across the five benchmarks, while Avg. Score denotes the mean of the reported benchmark scores.

β	Avg. TGR ↓	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Avg. Score ↑
∞	5.584	20.1	27.7	59.0	65.1	60.1	46.4
5.0	4.128	21.4	29.2	59.1	64.5	60.8	47.0
4.0	3.497	22.9	29.9	59.0	64.6	61.6	47.6
3.0	2.834	22.9	32.7	59.3	64.7	59.8	47.9
2.0	2.676	**24.9**	**37.9**	59.3	**66.3**	**61.8**	**50.0**
1.0	**2.675**	24.1	32.7	**59.5**	64.8	60.6	48.3

AudioFlamingo-Next-Instruct. Table 10 shows a similar trend on AudioFlamingo-Next-Instruct. Temporal Coverage Refinement progressively reduces Avg. TGR from 5.612 at $\beta = \infty$ to 3.266 at $\beta = 2$, while Avg. Score increases from 51.4 to 53.8. Further decreasing β to 1 yields only a marginal reduction in Avg. TGR to 3.261, while Avg. Score decreases to 52.6. Overall, Temporal Coverage Refinement effectively regulates excessive temporal sparsity while maintaining downstream performance across a different ALLM, with $\beta = 2$ achieving the highest Avg. Score among the evaluated settings.

Table 10: **Effect of Temporal Coverage Refinement on AudioFlamingo-Next-Instruct.** Results across five benchmarks at 90% pruning with Time-Sync. Deviation and cosine distance fixed. Avg. TGR denotes the mean TGR across the five benchmarks, while Avg. Score denotes the mean of the reported benchmark scores.

β	Avg. TGR ↓	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Avg. Score ↑
∞	5.612	42.9	47.4	30.9	67.3	68.5	51.4
5.0	4.409	42.8	51.8	30.4	67.4	71.1	52.7
4.0	3.988	43.1	51.9	29.2	67.9	70.9	52.6
3.0	3.472	43.4	51.7	31.3	67.5	70.5	52.9
2.0	3.266	**43.8**	**52.9**	**32.9**	**68.1**	**71.4**	**53.8**
1.0	**3.261**	41.9	51.0	31.3	67.8	71.0	52.6

MOSS-Audio-8B-Instruct. Table 11 shows that Temporal Coverage Refinement substantially reduces Avg. TGR on MOSS-Audio, from 3.925 at $\beta = \infty$ to 2.446 at $\beta = 2$, while Avg. Score remains relatively stable, ranging from 61.7 to 62.4 across the evaluated settings. The highest Avg. Score of 62.4 is achieved at $\beta = 3$. Further decreasing β from 2 to 1 yields only a marginal reduction in Avg. TGR from 2.446 to 2.443, while Avg. Score decreases from 62.2 to 62.0. Overall, Temporal Coverage Refinement effectively regulates excessive temporal sparsity while maintaining stable downstream performance on MOSS-Audio.

Table 11: **Effect of Temporal Coverage Refinement on MOSS-Audio-8B-Instruct.** Results across three benchmarks at 90% pruning with Time-Sync. Deviation and ℓ_2 distance fixed. Avg. TGR denotes the mean TGR across the three benchmarks, while Avg. Score denotes the mean of the reported benchmark scores.

β	Avg. TGR ↓	MELD	ClothoAQA	MMAU	Avg. Score ↑
∞	3.925	60.1	56.1	69.0	61.7
5.0	3.191	61.1	56.5	68.3	62.0
4.0	2.892	61.2	**56.6**	68.2	62.0
3.0	2.547	**61.5**	56.1	**69.5**	**62.4**
2.0	2.446	61.1	56.4	69.0	62.2
1.0	**2.443**	61.0	56.5	68.4	62.0

A.3 Cross-Model Analysis of Trajectory-Driven Refinement

We extend the refinement-criterion analysis presented in the main paper by providing results across a broader set of benchmarks and additional ALLMs with different audio representation pipelines. Following the same controlled setting, we compare Midpoint, Adjacent Change, and Time-Synchronized Deviation while keeping the ℓ_2 deviation metric and Temporal Coverage Refinement fixed. For Qwen2.5-Omni-7B, we additionally report results on ClothoAQA and AudioCaps beyond the representative benchmarks considered in the main ablation, and use the same benchmark set for the cross-model analysis. This controlled comparison examines whether the behavior of trajectory-driven refinement remains consistent across different audio representation spaces.

Qwen2.5-Omni-7B. Table 12 extends the refinement criterion analysis in the main paper to five benchmarks. Midpoint selects the center of each refinement interval, while Adjacent Change selects the token with the largest local representation change. Time-Sync. Deviation achieves the highest Avg. Score of 50.0, compared with 47.1 and 46.8 for Midpoint and Adjacent Change, respectively. It

also achieves the highest score on all five benchmarks, including a tie with Midpoint on ClothoAQA. These results further support trajectory-driven refinement across a broader set of downstream tasks.

Table 12: **Effect of the Refinement Criterion on Qwen2.5-Omni-7B.** Results across five benchmarks at 90% pruning with Temporal Coverage Refinement and the deviation metric fixed. Avg. Score denotes the mean of the reported benchmark scores.

Refinement	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Avg. Score
Midpoint	22.1	30.6	**59.3**	64.7	58.8	47.1
Adjacent Change	21.4	28.4	59.2	64.2	60.9	46.8
Time-Sync. Deviation	**24.9**	**37.9**	**59.3**	**66.3**	**61.8**	**50.0**

AudioFlamingo-Next-Instruct. Table 13 shows a similar trend on AudioFlamingo-Next-Instruct. Time-Sync. Deviation achieves the highest Avg. Score of 53.8, compared with 51.7 and 51.0 for Midpoint and Adjacent Change, respectively. It also achieves the highest score across all five benchmarks. These results further support trajectory-driven refinement across a different ALLM.

Table 13: **Effect of the Refinement Criterion on AudioFlamingo-Next-Instruct.** Results across five benchmarks at 90% pruning with Temporal Coverage Refinement and cosine distance fixed. Avg. Score denotes the mean of the reported benchmark scores.

Refinement	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Avg. Score
Midpoint	40.1	50.7	32.5	66.4	69.0	51.7
Adjacent Change	39.2	49.5	30.8	66.7	68.7	51.0
Time-Sync. Deviation	**43.8**	**52.9**	**32.9**	**68.1**	**71.4**	**53.8**

MOSS-Audio-8B-Instruct. Table 14 further evaluates the refinement criteria on MOSS-Audio. Time-Sync. Deviation achieves the highest Avg. Score of 62.4, compared with 61.5 and 61.1 for Midpoint and Adjacent Change, respectively. It also achieves the highest score across all three benchmarks. These results further support trajectory-driven refinement across a distinct audio representation pipeline.

Table 14: **Effect of the Refinement Criterion on MOSS-Audio-8B-Instruct.** Results across three benchmarks at 90% pruning with Temporal Coverage Refinement and ℓ_2 distance fixed. Avg. Score denotes the mean of the reported benchmark scores.

Refinement	MELD	ClothoAQA	MMAU	Avg. Score
Midpoint	61.0	55.3	68.2	61.5
Adjacent Change	61.2	55.1	66.9	61.1
Time-Sync. Deviation	**61.5**	**56.1**	**69.5**	**62.4**

A.4 Cross-Model Analysis of Representation Deviation Metrics

We extend the representation-deviation-metric analysis presented in the main paper by providing results across a broader set of benchmarks and additional ALLMs with different audio representation pipelines. We compare ℓ_1 distance, ℓ_2 distance, cosine distance, and negative dot product while keeping the time-synchronized deviation criterion and Temporal Coverage Refinement fixed. For Qwen2.5-Omni-7B, we additionally report results on ClothoAQA and AudioCaps beyond the representative benchmarks considered in the main ablation, and use the same benchmark set for the cross-model analysis. This controlled comparison examines how the choice of representation deviation metric affects Time-Aware RDP across different audio representation spaces.

Qwen2.5-Omni-7B. Table 15 extends the deviation metric analysis in the main paper to five benchmarks. The four deviation metrics exhibit relatively similar performance across the evaluated tasks, with Cosine Distance achieving the highest Avg. Score of 50.0, followed by Negative Dot Product, ℓ_1 Distance, and ℓ_2 Distance with 49.1, 48.2, and 48.1, respectively. No single metric consistently performs best across all benchmarks, indicating that trajectory-driven refinement remains effective across different choices of representation deviation metric.

Table 15: **Effect of the Deviation Metric on Qwen2.5-Omni-7B.** Results across five benchmarks at 90% pruning with Temporal Coverage Refinement and the refinement criterion fixed. Avg. Score denotes the mean of the reported benchmark scores.

Deviation Metric	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Avg. Score
ℓ_1 Distance	24.1	31.2	58.7	65.5	61.6	48.2
ℓ_2 Distance	22.6	32.8	**59.4**	64.8	60.7	48.1
Cosine Distance	**24.9**	**37.9**	59.3	**66.3**	**61.8**	**50.0**
Negative Dot Product	22.9	**37.9**	59.0	66.0	59.9	49.1

AudioFlamingo-Next-Instruct. Table 16 further evaluates the effect of the representation deviation metric on AudioFlamingo-Next-Instruct. The four deviation metrics show comparable performance across the evaluated tasks, with Cosine Distance achieving the highest Avg. Score of 52.9, followed closely by Negative Dot Product at 52.8. The ℓ_2 and ℓ_1 distances achieve Avg. Scores of 51.7 and 51.3, respectively. No single metric consistently performs best across all benchmarks, further indicating that trajectory-driven refinement remains effective across different choices of representation deviation metric.

Table 16: **Effect of the Deviation Metric on AudioFlamingo-Next-Instruct.** Results across five benchmarks at 90% pruning with Temporal Coverage Refinement and the refinement criterion fixed. Avg. Score denotes the mean of the reported benchmark scores.

Deviation Metric	Clotho v2	AudioCaps	MELD	ClothoAQA	MMAU	Avg. Score
ℓ_1 Distance	42.5	47.8	29.6	67.4	69.2	51.3
ℓ_2 Distance	43.6	47.4	29.7	67.8	69.9	51.7
Cosine Distance	**43.8**	**52.9**	**32.9**	**68.1**	**71.4**	**53.8**
Negative Dot Product	41.3	51.1	**32.9**	67.9	70.8	52.8

MOSS-Audio-8B-Instruct. Table 17 further evaluates the effect of the representation deviation metric on MOSS-Audio. The ℓ_2 distance achieves the highest Avg. Score of 62.4, followed by ℓ_1 distance at 61.8, while Cosine Distance and Negative Dot Product achieve 61.3 and 61.1, respectively. Overall, the results show that trajectory-driven refinement remains effective across different deviation metrics, with ℓ_2 distance providing the strongest performance on MOSS-Audio.

Table 17: **Effect of the Deviation Metric on MOSS-Audio-8B-Instruct.** Results across three benchmarks at 90% pruning with Temporal Coverage Refinement and the refinement criterion fixed. Avg. Score denotes the mean of the reported benchmark scores.

Deviation Metric	MELD	ClothoAQA	MMAU	Avg. Score
ℓ_1 Distance	61.3	**56.1**	68.1	61.8
ℓ_2 Distance	**61.5**	**56.1**	**69.5**	**62.4**
Cosine Distance	61.1	54.5	68.4	61.3
Negative Dot Product	61.1	54.3	68.0	61.1

B Detailed Experimental Setup

B.1 Models

We evaluate TimePrune on four ALLMs with different model architectures and audio representation pipelines: Qwen2.5-Omni-7B (Xu et al., 2025a), AudioFlamingo-Next-Instruct (Ghosh et al., 2026), Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025b), and MOSS-Audio-8B-Instruct (Yang et al., 2026). For all models, TimePrune operates on temporally ordered audio representations produced by the corresponding audio encoder without additional training or modification of model parameters. Given a pruning ratio R and N audio tokens, we retain $K = \max(2, \lfloor (1-R)N \rfloor)$ temporal positions, with the two endpoints always retained. The selected temporal positions are

propagated through the model-specific audio processing pipeline while preserving the original inference procedure.

Qwen2.5-Omni-7B. We use the official Qwen2.5-Omni implementation[1] as the primary model for the main analysis and controlled ablation studies. Audio is processed by the Thinker audio tower, where consecutive encoder frames are pair-pooled to form a temporal sequence aligned one-to-one with the corresponding audio placeholders. TimePrune selects temporal positions from this encoder sequence, and the same positions are retained in the corresponding post-projector audio representations. Unused audio placeholders are removed accordingly, after which generation follows the original Thinker inference pipeline with the official chat template and greedy decoding.

AudioFlamingo-Next-Instruct. We use the official Audio Flamingo implementation[2]. Its audio front-end processes 16 kHz audio in 30-second windows, with empty leftover windows discarded following the model's audio processing pipeline. TimePrune selects temporal positions from the post-pooling encoder representations before rotary time embeddings and multimodal projection. The selected positions are propagated to the corresponding projected audio representations, which are then provided to the language model through the remaining audio placeholders. All subsequent processing follows the original inference pipeline.

Qwen3-Omni-30B-A3B-Instruct. We use the official Qwen3-Omni implementation[3]. Unlike Qwen2.5-Omni, its audio tower downsamples the input through stacked convolutions without an additional pair-pooling stage. TimePrune selects temporal positions from the resulting encoder representations before projection, and the same positions are retained in the corresponding projected audio representations and audio placeholders. The compressed multimodal input is then processed through the original MoE Thinker inference pipeline without modifying the audio tower, expert routing, or trained parameters.

MOSS-Audio-8B-Instruct. We use the official MOSS-Audio implementation[4]. MOSS-Audio propagates audio information through both the final encoder stream and DeepStack intermediate features. TimePrune selects temporal positions from the final encoder representations before the audio adapter, and the same positions are retained across the encoder representations, DeepStack features, and corresponding audio placeholders to preserve temporal alignment. The pruned streams are subsequently processed by the original audio adapter and DeepStack merging operations, while time-marker and text tokens remain unchanged.

B.2 Datasets and Evaluation Protocols

We evaluate TimePrune on five audio understanding benchmarks covering audio captioning, speech emotion recognition, audio question answering, and comprehensive audio reasoning. Unless otherwise specified, we follow the official dataset splits and evaluate each model using its original inference pipeline and chat template. We use greedy decoding for all experiments.

Clotho v2 (Drossos et al., 2020). Clotho is an audio captioning dataset containing audio clips paired with multiple human-written captions. We evaluate on the official evaluation split using the reference captions provided by the dataset and report CIDEr multiplied by 100. We use the official Clotho dataset release[5].

AudioCaps (Kim et al., 2019). AudioCaps is an audio captioning dataset containing audio clips paired with natural-language descriptions of their acoustic content. We evaluate on the official test split using the reference captions and report CIDEr multiplied by 100. We use the official AudioCaps release[6].

MELD (Poria et al., 2019). MELD is a multimodal conversational emotion recognition dataset containing utterances annotated with seven emotion categories: anger, disgust, fear, joy, neutral, sadness, and surprise. We evaluate speech emotion recognition on the official test split and report

[1] https://github.com/QwenLM/Qwen2.5-Omni
[2] https://github.com/NVIDIA/audio-flamingo
[3] https://github.com/QwenLM/Qwen3-Omni
[4] https://github.com/OpenMOSS/MOSS-Audio
[5] https://github.com/audio-captioning/clotho-dataset.
[6] https://github.com/cdjkim/audiocaps

classification accuracy. For all four models, each sample is evaluated using the audio signal together with its corresponding gold utterance text, and the model predicts one of the seven emotion categories. The same input protocol is used across Vanilla and all token compression methods to ensure that comparisons differ only in audio token compression. We use the official MELD release[7].

ClothoAQA (Lipping et al., 2022). Clotho-AQA is an audio question answering benchmark constructed from Clotho audio clips, where each audio sample is paired with multiple natural-language questions and answers. The benchmark contains both binary yes/no questions and questions with single-word answers. We evaluate on the official test split and report accuracy over the combined benchmark. The main results report the combined ClothoAQA accuracy, while the detailed results separately report performance by question type. We use the official Clotho-AQA release[8].

MMAU (Sakshi et al., 2025). MMAU is a comprehensive audio understanding and reasoning benchmark spanning sound, speech, and music. We evaluate on the publicly available 1,000-sample `test-mini` split. For the main results, we report the mean of the Sound, Speech, and Music scores after rounding each category score to one decimal place, while the detailed results report the three category scores separately. We use the official MMAU benchmark and evaluation code[9].

B.3 Compared Methods

We compare TimePrune with representative training-free token compression methods that use different sources of information for token reduction, including attention- and task-based importance, redundancy, temporal structure, diversity, and representativeness. All methods are evaluated using the same model, dataset, decoding protocol, and target pruning ratio R, while retaining their method-specific token-budget and selection rules. For methods originally developed for visual or multimodal token compression, we preserve the relevant compression principle while adapting the token-level operations to the audio modality. Depending on the method, token selection is performed on either encoder-side audio representations or representations after the model-specific audio projection or adaptation stage. No additional training is performed for these adaptations.

VisionZip (Yang et al., 2025)[10]. VisionZip combines attention-based importance with representation redundancy by identifying dominant tokens and incorporating information from contextual tokens through merging. In our audio adaptation, we apply this principle to encoder-side audio representations and their attention statistics, using a 54:10 ratio between dominant and contextual tokens. Contextual tokens are incorporated through the original ToMe-style merging procedure, and the resulting compressed representations are propagated through the remaining model pipeline.

DivPrune (Alvar et al., 2025)[11]. DivPrune formulates token selection from the perspective of representational diversity and greedily constructs a diverse subset using cosine distance. In our audio adaptation, we apply its greedy max-min selection criterion to the projected audio representations. Following the implementation used in our experiments, the number of retained tokens is determined as $\mathrm{round}(N(1-R))$.

FLoC (Cho et al., 2026b). FLoC formulates token compression as a facility-location problem and uses lazy-greedy subset selection to identify representative tokens that provide broad coverage of the original token set. In our audio adaptation, we apply the facility-location objective to the projected audio representations over the full temporal sequence (`block_size`=0), rather than partitioning the sequence into fixed-size blocks, and perform selection until the target token count is reached.

SpeechPrune (Lin et al., 2025). SpeechPrune performs speech token reduction through a two-phase procedure that combines frame-level selection with text-conditioned attention. We apply the method to the projected audio representations, using the input question together with the query and key projections from the first language-model layer to compute the text–audio attention used for token selection. The two phases share the pruning budget corresponding to the target ratio R, and the selected tokens are restored to their original temporal order before subsequent processing. We

[7] `https://github.com/declare-lab/MELD`
[8] `https://zenodo.org/records/6473207`.
[9] `https://github.com/Sakshi113/MMAU`.
[10] `https://github.com/JIA-Lab-research/VisionZip`
[11] `https://github.com/vbdi/divprune`

use 25 audio tokens per second for Qwen2.5-Omni and AudioFlamingo-Next-Instruct and 12 for MOSS-Audio, according to the temporal resolution of their respective audio representations.

Segmentwise Top-K (Gibier et al., 2026). Segmentwise Top-K incorporates temporal locality by partitioning the audio representation sequence into non-overlapping temporal segments and performing attention-based selection within each segment. We use $S = 10$ segments and compute token importance from encoder-side attention. Given a target of K retained tokens, each segment retains $\lfloor K/S \rfloor$ tokens according to its local importance scores.

Frame (He et al., 2025)[12]. We use the frame-level subsampling strategy provided in AudioMarathon as a temporal sampling baseline. Following the released implementation, given an encoder sequence of length N and a target of K retained tokens, the i-th retained position is determined by $\lfloor iN/K \rfloor$. The resulting positions are selected directly from the temporally ordered encoder representations without additional token scoring.

HeadRouter (He et al., 2026)[13]. HeadRouter performs task-adaptive audio token pruning using attention-head statistics obtained through offline calibration. We use the calibrated layer-1 head profiles derived from the AudioMarathon calibration data rather than recalibrating them on the evaluation benchmarks. During inference, per-head audio importance is computed from attention between textual queries and keys derived from the projected audio representations, and the calibrated head profiles are used to construct the final routing scores. Following the implementation used in our experiments, the retained token count is $\max(1, \mathrm{int}(N(1-R)))$.

OmniFit (Wang et al., 2026). OmniFit is originally designed for adaptive token compression across multiple modalities and layers. For our audio-only adaptation, we use its audio-side compression mechanism rather than the full cross-modal and layer-adaptive compression pipeline. Specifically, we compute the ℓ_2 norm of each post-adaptor audio representation, retain the highest-scoring tokens under the target pruning ratio, and apply the merging procedure used in its audio compression stage. Cross-modal opposing signals and in-LLM layer-adaptive compression are not used in this comparison.

OmniZip (Tao et al., 2026). OmniZip is originally designed for audio-guided multimodal token compression. For our audio-only adaptation, we use its audio compression branch without visual features rather than the full multimodal compression pipeline. Audio token scores are computed from encoder-side audio representations and attention, followed by its audio selection and contextual-token processing. Because the procedure additionally retains contextual tokens, we adjust the merging ratio in the main comparison to account for the 3% contextual-token allocation and approximately match the target pruning ratio R. For MOSS-Audio, we use the corresponding audio compression setting without this budget-matching adjustment.

C Additional Related Work

C.1 Token Pruning and Compression

Audio Token Compression. Audio Large Language Models (ALLMs) have been developed to support diverse speech, sound, and music understanding tasks, with representative models including Qwen-Audio (Chu et al. (2023)) and Qwen2-Audio (Chu et al. (2024)), the Audio Flamingo series (Kong et al. (2024); Ghosh et al. (2025); Goel et al. (2025); Ghosh et al. (2026)), MOSS-Audio (Yang et al. (2026)), SALMONN (Tang et al. (2024)), Kimi-Audio (KimiTeam et al. (2025)), and MiMo-Audio (Xiaomi LLM-Core Team et al. (2025)). As these models process temporally dense audio representations, recent training-free approaches have explored audio token compression using attention- and task-based importance, representation redundancy, and temporal structure. SpeechPrune (Lin et al. (2025)) combines speech–text similarity with approximated attention to identify context-relevant speech tokens, while Segmentwise Top-K (Gibier et al. (2026)) performs token selection within temporal segments to mitigate the temporal imbalance induced by global pruning. HeadRouter (He et al. (2026)) further exploits task-dependent attention-head behavior obtained through offline calibration for adaptive importance estimation, whereas LTBM (Luo et al. (2026)) reduces redundancy by merging representationally similar tokens within local tem-

[12] https://github.com/DabDans/AudioMarathon

[13] https://github.com/DabDans/Headrouter_arxiv

poral neighborhoods. These methods incorporate importance, redundancy, or temporal information into audio token compression, but their retained temporal structure is primarily determined by token importance, predefined temporal locality, or representation similarity. In contrast, TimePrune directly evaluates whether the retained anchors are sufficient to approximate the underlying temporal representation trajectory and recursively introduces additional anchors where this approximation is insufficient.

Visual Token Compression. Token reduction has been extensively studied in VLMs such as LLaVA (Liu et al. (2023)), LLaVA-OneVision (Li et al. (2024a)), and Qwen2.5-VL (Bai et al. (2025)). Existing approaches construct compact visual representations using attention and saliency, as in FastV (Chen et al. (2024)), SparseVLM (Zhang et al. (2025c)), FasterVLM (Zhang et al. (2024)), and VisPruner (Zhang et al. (2025a)); feature similarity and redundancy, as in VisionZip (Yang et al. (2025)) and DART (Wen et al. (2025)); and representation diversity or coverage, as in DivPrune (Alvar et al. (2025)), CDPruner (Zhang et al. (2025b)), and SCOPE (Deng et al. (2025)). More recent methods incorporate token sensitivity, representation transitions, and joint salience–coverage objectives, including ZOO-Prune (Kim et al. (2026)), TransPrune (Li et al. (2026)), and SCoRe (Xu et al. (2026)), while broader subset-selection formulations such as FLoC (Cho et al. (2026b)) and MMTok (Dong et al. (2026)) aim to preserve representative or multimodally informative tokens. For temporally structured visual inputs, FlashVID (Fan et al. (2026)) combines token selection with spatiotemporal merging to exploit redundancy across spatial and temporal dimensions, whereas SURGE (Tang et al. (2026)) exploits temporal predictability for token retention. Complementary studies such as RESTORE (Cho et al. (2026a)) and LiteLVLM (Lee & Choi (2026)) further investigate positional distortions induced by token compression and task-specific token reduction. While these approaches provide effective criteria for identifying informative and representative tokens, including for temporally structured visual inputs, they do not explicitly assess whether the retained tokens provide sufficient temporal resolution to approximate an underlying representation trajectory.

Multimodal Token Compression. Multimodal Large Language Models (MLLMs) have been developed to jointly process multiple modalities, with representative models including Baichuan-Omni (Li et al. (2024b)), Qwen2.5-Omni (Xu et al. (2025a)), Ming-Omni (Inclusion AI (2025)), Qwen3-Omni (Xu et al. (2025b)), and MiniCPM-o 4.5 (Cui et al. (2026)). Along with these models, token compression has expanded toward coordinated reduction across multiple modalities by exploiting cross-modal correspondence and multimodal interactions. OmniZip (Tao et al. (2026)) uses audio saliency and cross-modal similarity for audio-guided multimodal compression, OmniSIFT (Ding et al. (2026)) performs modality-asymmetric visual compression followed by vision-guided audio selection, and OmniFit (Wang et al. (2026)) adaptively allocates compression across layers and modalities according to redundancy and cross-modal alignment. Recent approaches further explore different multimodal compression strategies: DASH (Li & Huang (2026)) leverages audio-driven temporal segmentation, SEATS (Xin et al. (2026)) performs stage-adaptive token selection, Omni-Prune (Zhong et al. (2026)) incorporates query-aware multimodal information, and OmniPack (Su et al. (2026)) jointly compresses tokens across modalities. While these approaches demonstrate that interactions among modalities can provide useful signals for token compression, their audio token allocation is generally conditioned on saliency, redundancy, segmentation, or information from other modalities. TimePrune instead derives temporal resolution directly from the geometry of the audio representation trajectory and therefore does not require an accompanying visual stream or cross-modal correspondence for token selection.

C.2 Trajectory Simplification and RDP

The Ramer–Douglas–Peucker (RDP) algorithm recursively simplifies geometric trajectories by retaining points with large approximation deviations Ramer (1972); Douglas & Peucker (1973). Given a trajectory and its endpoints, RDP identifies the point with the largest deviation from the approximating segment and recursively refines the resulting sub-trajectories. This process produces a compact set of representative points while preserving the geometric structure of the original trajectory.

Subsequent trajectory simplification studies extend this principle by explicitly incorporating temporal information. In particular, time-synchronized trajectory simplification measures the deviation between an observation and its temporally corresponding interpolated position on the simplified trajectory, thereby accounting for both spatial and temporal information Meratnia & de By (2004);

Lin et al. (2019); Tamilmani & Stefanakis (2019). Unlike conventional geometric deviation, the interpolated position is determined according to the relative timestamp of the observation between the two retained endpoints. SED-based trajectory simplification has also been studied for efficient error-bounded compression, demonstrating the broader applicability of time-synchronized deviation in trajectory reduction Lin et al. (2019). These studies establish time-aware trajectory approximation as an effective mechanism for preserving trajectory structure with a reduced number of points.

TimePrune builds upon this trajectory-simplification principle but considers a different application setting: training-free compression of high-dimensional audio representations under a fixed token budget for Audio Large Language Models. Rather than treating trajectory simplification as the final objective, TimePrune uses time-synchronized representation deviation to determine where additional temporal resolution is required. Starting from sparse temporal anchors, it recursively refines intervals whose retained anchors insufficiently approximate the original representation trajectory, naturally producing non-uniform temporal resolution without explicitly defining local complexity scores or temporal segments. Moreover, conventional RDP-style simplification primarily focuses on approximation fidelity, whereas TimePrune additionally incorporates Temporal Coverage Refinement to prioritize excessively sparse temporal intervals under a limited token budget. Thus, TimePrune adapts trajectory simplification to audio token compression by jointly considering trajectory-driven adaptive temporal resolution and temporal coverage.

D Additional Analysis of Temporal Trajectory Preservation

D.1 Effect of Temporal Coverage Refinement across Pruning Ratios

We further analyze the effect of Temporal Coverage Refinement across different token budgets. Figure 6 compares TimePrune with $\beta \in \{2, 3\}$ and Time-Aware RDP without coverage refinement ($\beta = \infty$) at pruning ratios of 75%, 85%, and 90% across five datasets. We report SED and TGR as complementary diagnostics of trajectory fidelity and temporal coverage, respectively.

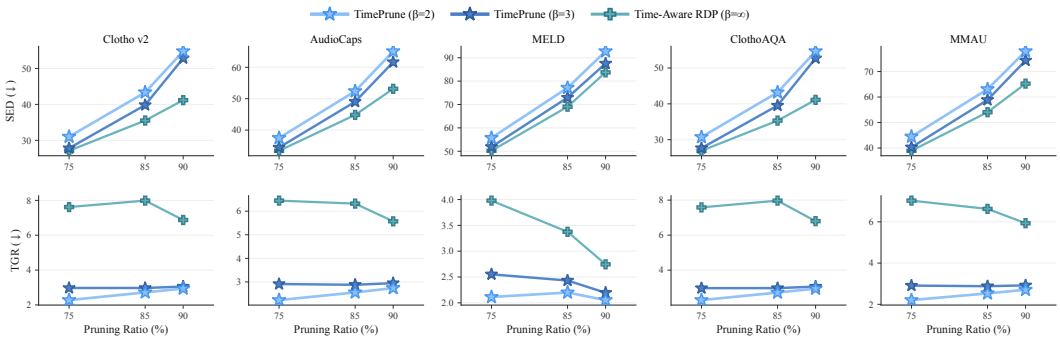
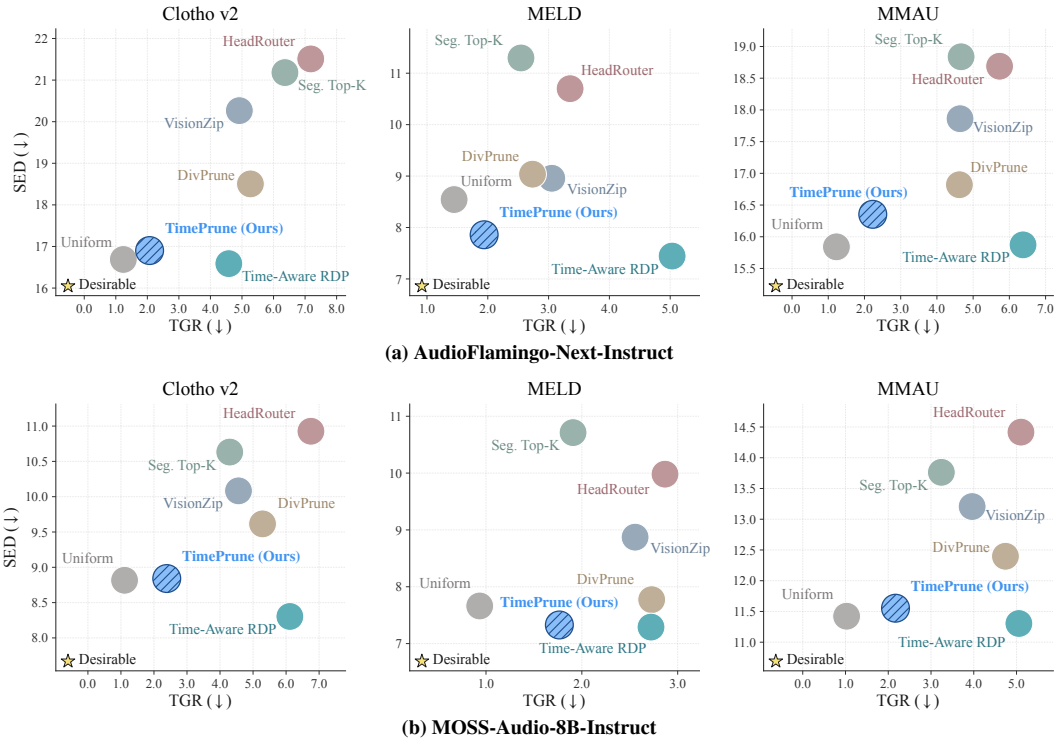


Figure 6: **Effect of Temporal Coverage Refinement across pruning ratios and datasets.** We compare TimePrune with $\beta \in \{2, 3\}$ and Time-Aware RDP without Temporal Coverage Refinement ($\beta = \infty$) on Qwen2.5-Omni-7B. Each column reports SED (top) and TGR (bottom) at pruning ratios of 75%, 85%, and 90%. Lower is better for both metrics.

As shown in Figure 6, Time-Aware RDP consistently achieves lower SED but exhibits substantially higher TGR, indicating more excessive temporal sparsity. Introducing Temporal Coverage Refinement substantially reduces TGR across datasets and pruning ratios while yielding moderately higher SED, reflecting the trade-off introduced by prioritizing excessively sparse intervals during refinement. The two finite settings, $\beta = 2$ and $\beta = 3$, exhibit broadly similar coverage behavior compared with the substantially higher TGR observed at $\beta = \infty$. Overall, these results show that Temporal Coverage Refinement consistently regulates excessive temporal sparsity across different token budgets while retaining the trajectory-driven behavior of Time-Aware RDP.

D.2 Cross-Method Analysis of Trajectory Fidelity and Temporal Coverage

We extend the trajectory fidelity and temporal coverage analysis in the main paper to different audio representation pipelines. Figure 7 compares different pruning methods on AudioFlamingo-Next-Instruct and MOSS-Audio-8B-Instruct using SED and TGR. Lower SED indicates a more faithful piecewise approximation of the original representation trajectory, whereas lower TGR indicates fewer excessively sparse temporal intervals.



(a) AudioFlamingo-Next-Instruct

(b) MOSS-Audio-8B-Instruct

Figure 7: **Cross-model and cross-method analysis of trajectory fidelity and temporal coverage.** We compare different token pruning methods in terms of SED and TGR on Clotho-v2, MELD, and MMAU using AudioFlamingo-Next-Instruct and MOSS-Audio-8B-Instruct. Lower values indicate better trajectory fidelity (SED) and temporal coverage (TGR), respectively. SED values are interpreted within each model's representation space and are not directly comparable across models.

Across both models, the pruning methods exhibit different fidelity–coverage characteristics, with their relative positions varying across datasets and representation spaces. Uniform retention generally exhibits low TGR due to its regular temporal spacing, while Time-Aware RDP achieves low SED but substantially higher TGR. TimePrune reduces the high TGR of Time-Aware RDP while maintaining relatively low SED. These results extend the observation from the main paper to different audio representation pipelines, further showing that trajectory fidelity and temporal coverage provide complementary views of the retained temporal structure.

D.3 Visualization of Temporal Trajectory Preservation

Figure 8 provides qualitative examples of the retained representation trajectories for different pruning methods on Clotho v2, MELD, and MMAU. For each sample, we visualize the original trajectory together with the trajectories reconstructed from the retained temporal anchors under the same token budget. To visualize the high-dimensional encoder representations, we project each sample independently onto its first two principal components and construct a three-dimensional trajectory using *Time* $\times$ *PC1* $\times$ *PC2*, where the time axis corresponds to the audio-token index $t = 0, \ldots, T-1$. For visualization only, the trajectory is smoothed using a moving average with a window size of 11, and the Original trajectory is displayed using at most 40 uniformly sampled points.

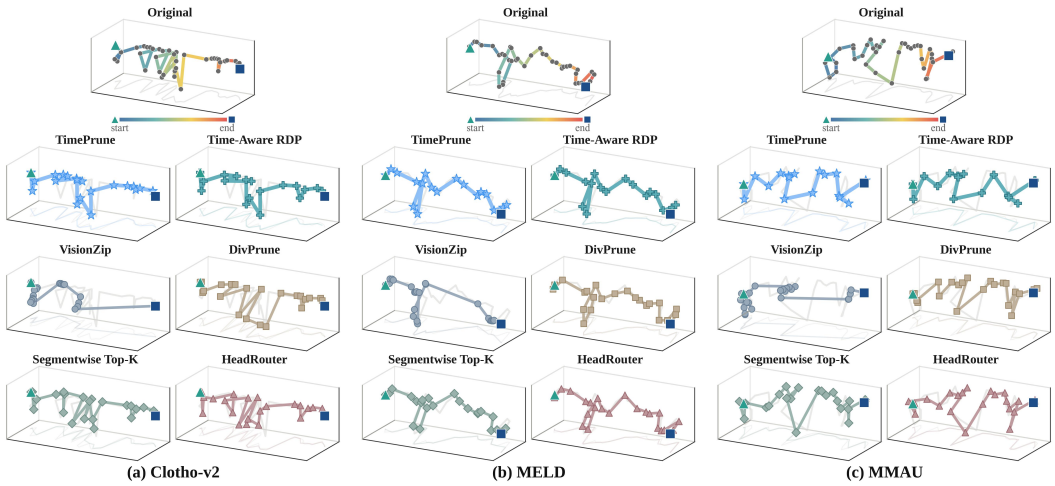


Figure 8: **Visualization of temporal trajectory preservation.** Original and retained representation trajectories for samples from (a) Clotho v2, (b) MELD, and (c) MMAU under the same token budget. Each trajectory is visualized in the Time $\times$ PC1 $\times$ PC2 space. Green triangles and blue squares indicate the start and end of each trajectory, respectively, and the color bar below the Original trajectory indicates temporal progression from start to end.

As shown in Fig. 8, different pruning strategies retain substantially different sets of temporal anchors, resulting in distinct approximations of the original representation trajectory. Time-Aware RDP follows the original trajectory through trajectory-driven refinement, while TimePrune additionally incorporates Temporal Coverage Refinement to regulate excessively sparse temporal intervals. These visualizations complement the quantitative SED and TGR analysis in Sec. D.2 by providing a qualitative view of how different anchor-selection strategies preserve the temporal representation trajectory.

D.4 Additional Temporal Coverage Analysis

TGR summarizes the largest inter-anchor gap relative to the nominal uniform spacing but does not characterize the full distribution of temporal gaps. We therefore complement the TGR analysis by considering the normalized gap between consecutive retained anchors,

$$\tilde{g}_i = \frac{a_{i+1} - a_i}{(T-1)/(K-1)}, \tag{8}$$

where $\tilde{g}_i = 1$ corresponds to the nominal uniform spacing under the same sequence length and token budget. To characterize the distribution of these gaps, we plot their complementary cumulative distribution function (CCDF),

$$\mathrm{CCDF}(g) = \frac{1}{M} \sum_{i=1}^{M} \mathbf{1}[\tilde{g}_i \geq g], \tag{9}$$

where M is the total number of inter-anchor gaps. Thus, the horizontal axis represents the normalized gap g, while $\mathrm{CCDF}(g)$ represents the fraction of inter-anchor gaps whose normalized length is at least g.

As shown in Fig. 9, Time-Aware RDP without coverage refinement ($\beta = \infty$) exhibits a longer upper tail, indicating the presence of substantially larger inter-anchor gaps relative to the nominal spacing. Temporal Coverage Refinement ($\beta \in \{2, 3\}$) reduces this upper tail across the evaluated settings while preserving a non-uniform distribution of temporal gaps. These results further show that Temporal Coverage Refinement regulates excessively sparse temporal intervals rather than enforcing uniform spacing.

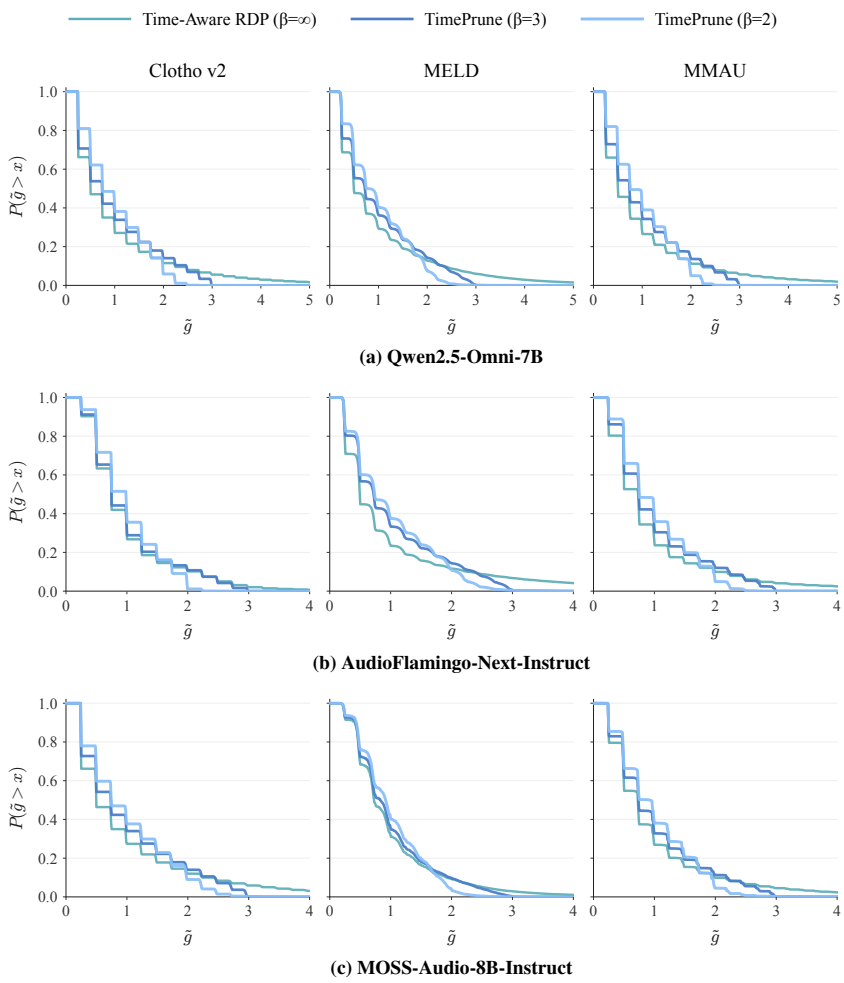


Figure 9: **Distribution of normalized inter-anchor gaps across model architectures.** Complementary cumulative distribution functions (CCDFs) of normalized gaps between consecutive retained anchors on Clotho-v2, MELD, and MMAU at 75% pruning. We compare Time-Aware RDP without Temporal Coverage Refinement ($\beta = \infty$) with TimePrune using $\beta \in \{2, 3\}$.

E Additional Analysis of Adaptive Temporal Resolution

In this section, we provide additional analyses of the adaptive temporal resolution induced by trajectory-driven refinement. While the main paper shows that Time-Aware RDP naturally allocates relatively denser temporal anchors to regions with higher local representation variation, we further examine this behavior from two complementary perspectives. We first compare the temporal allocation patterns of TimePrune, Time-Aware RDP, and representative pruning methods. We then analyze how Temporal Coverage Refinement affects the adaptive allocation induced by trajectory-driven refinement.

E.1 Cross-Method Analysis of Adaptive Temporal Allocation

We extend the analysis in Sec. 2.2 by comparing the temporal allocation patterns of TimePrune, Time-Aware RDP, and representative pruning methods across different audio representation pipelines. Following the main analysis, we divide each audio sequence into equal temporal regions, compute the local representation variation of each region, and group the regions into five variation quantiles from Q1 to Q5. We then measure the retained token density within each quantile under the same token budget.

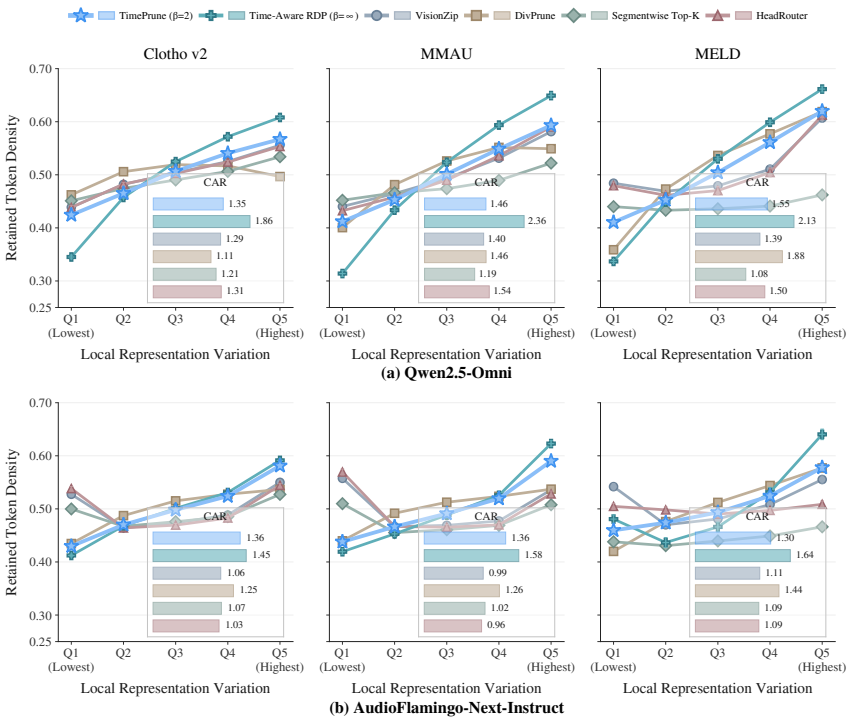


Figure 10: **Cross-method analysis of adaptive temporal allocation.** Retained token density across local representation-variation quantiles (Q1–Q5) on Clotho v2, MMAU, and MELD at 50% pruning. We compare TimePrune ($\beta = 2$), Time-Aware RDP ($\beta = \infty$), and representative pruning methods on Qwen2.5-Omni and AudioFlamingo-Next-Instruct. CAR $= D_{Q5}/D_{Q1}$ measures the relative allocation between the highest- and lowest-variation regions.

As shown in Fig. 10, Time-Aware RDP consistently increases retained token density toward higher-variation regions across both model architectures and all three datasets, despite not explicitly using local representation variation for token selection. Incorporating Temporal Coverage Refinement moderates this variation-dependent allocation, with TimePrune exhibiting lower CAR than Time-Aware RDP while maintaining a non-uniform allocation across variation levels. The other pruning methods show more model- and dataset-dependent allocation patterns. These results further support that trajectory-driven refinement naturally induces adaptive temporal resolution across different audio representation pipelines, while Temporal Coverage Refinement moderates this allocation by additionally prioritizing excessively sparse temporal intervals.

E.2 Effect of Temporal Coverage Refinement on Token Allocation

We next examine how Temporal Coverage Refinement affects the adaptive temporal allocation induced by trajectory-driven refinement. We compare Time-Aware RDP ($\beta = \infty$), which does not use coverage refinement, with TimePrune using $\beta \in \{3, 2\}$. All variants use the same time-synchronized representation deviation criterion and token budget, differing only in the strength of Temporal Coverage Refinement.

As shown in Fig. 11, Time-Aware RDP ($\beta = \infty$) exhibits the strongest variation-dependent allocation, with retained token density generally increasing toward higher-variation regions. Introducing Temporal Coverage Refinement progressively moderates this allocation as β decreases. Across both model architectures and all three datasets, CAR consistently decreases from $\beta = \infty$ to $\beta = 3$ and further to $\beta = 2$. Nevertheless, TimePrune maintains CAR above 1 in all settings, indicating that the resulting allocation remains non-uniform with relatively higher retained density in higher-variation regions. Together with the temporal coverage analysis in Sec. D.4, these results show that Temporal Coverage Refinement regulates excessive temporal sparsity while moderating, rather than eliminating, the adaptive temporal allocation induced by trajectory-driven refinement.

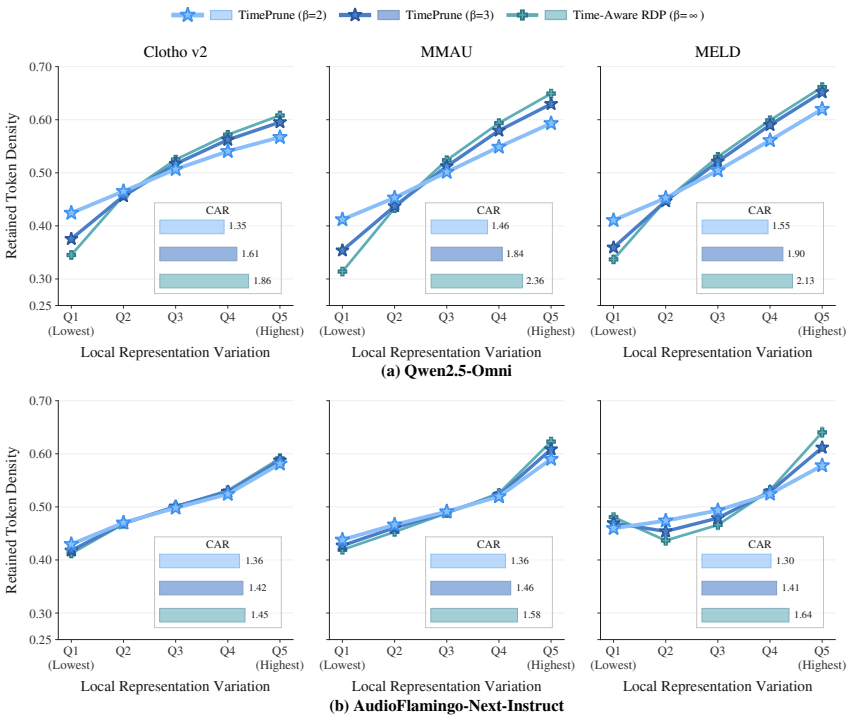


Figure 11: **Effect of Temporal Coverage Refinement on adaptive temporal allocation.** Retained token density across local representation-variation quantiles (Q1–Q5) on Clotho v2, MMAU, and MELD at 50% pruning. We compare Time-Aware RDP ($\beta = \infty$) with TimePrune using $\beta \in \{3, 2\}$ on Qwen2.5-Omni and AudioFlamingo-Next-Instruct. $\mathrm{CAR} = D_{Q5}/D_{Q1}$ measures the relative allocation between the highest- and lowest-variation regions.

E.3 Sample-Level Visualization of Adaptive Temporal Allocation

We further provide sample-level visualizations to illustrate how temporal token allocation varies within individual audio sequences. Figure 12 presents examples from Clotho v2, MELD, and MMAU, showing the waveform, regional representation variation, and retained token allocation over time. All methods are compared under the same token budget, with TimePrune using $\beta = 2$.

As shown in Fig. 12, the compared methods produce distinct temporal allocation patterns within individual audio sequences. Time-Aware RDP exhibits substantial variation in local token density over time, reflecting its trajectory-driven refinement. TimePrune moderates these large variations through Temporal Coverage Refinement while preserving non-uniform temporal allocation. Notably, the retained-token patterns do not directly follow regional representation variation, consistent with our formulation in which local variation is used only for analysis rather than as a token-selection criterion. These examples complement the aggregate analyses in Secs. E.1 and E.2 by illustrating their temporal allocation behavior at the sample level.

F Additional Efficiency Analysis

F.1 Efficiency Measurement Details

We provide additional details on the efficiency measurements reported in Sec. 4.4. We separately report theoretical computational cost and measured prefilling latency to distinguish the reduction in LLM computation from the overhead introduced by token compression. All methods are evaluated under the same inference configuration, and audio encoding time is excluded because token compression is applied to the encoded audio representations.

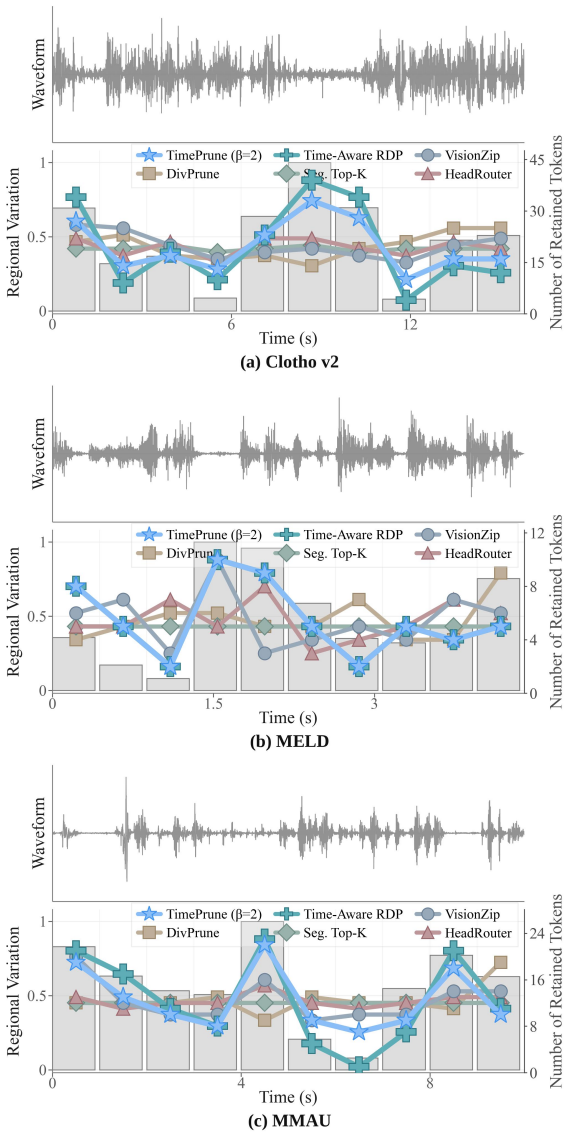


Figure 12: **Sample-level visualization of adaptive temporal allocation.** Regional representation variation and retained token allocation over time for samples from (a) Clotho v2, (b) MELD, and (c) MMAU. Bars indicate regional representation variation, while lines show the number of retained tokens within each temporal region.

FLOPs. We estimate the theoretical audio-token-dependent FLOPs of the LLM backbone during prefilling. The LLM backbones of all three evaluated models employ grouped-query attention (GQA) and SwiGLU feed-forward layers, allowing us to use a common FLOPs formulation while accounting for model-specific architectural differences through their respective parameters. Let n denote the mean number of retained audio tokens, d the hidden dimension, d_{ff} the intermediate dimension, L the number of Transformer layers, h the number of query heads, and h_{kv} the number of key–value heads. We compute

$$F_{\text{attn}} = 2nd^2 \left(1 + \frac{h_{\text{kv}}}{h}\right) + 2n^2 d, \tag{10}$$

$$F_{\text{ffn}} = 3nd\, d_{\text{ff}}, \tag{11}$$

$$F_{\text{LLM}} = L\left(F_{\text{attn}} + F_{\text{ffn}}\right). \tag{12}$$

For Qwen2.5-Omni-7B and AudioFlamingo-Next-Instruct, we use $L = 28$, $d = 3584$, $d_{\text{ff}} = 18944$, $h = 28$, and $h_{\text{kv}} = 4$. For MOSS-Audio-8B-Instruct, we use $L = 36$, $d = 4096$, $d_{\text{ff}} = 12288$, $h = 32$, and $h_{\text{kv}} = 8$. For each model and compression method, we substitute the corresponding model-specific architectural parameters and the mean number of retained audio tokens into the same formulation.

Prefilling Time. We measure prefilling latency using a consistent protocol across all three evaluated models. We decompose the latency into *Compression* and *LLM Forward*. Compression measures the time required to select the retained audio tokens and construct the compressed input, excluding audio encoding. LLM Forward measures the subsequent LLM prefilling computation on the compressed sequence, including the first-token argmax operation. Accordingly, the total prefilling time is computed as

$$T_{\text{total}} = T_{\text{compression}} + T_{\text{LLM Forward}}. \tag{13}$$

We compute the prefilling speedup relative to Vanilla as

$$\text{Speedup} = \frac{T_{\text{total}}^{\text{Vanilla}}}{T_{\text{total}}}. \tag{14}$$

All latency measurements are conducted using the same hardware, a batch size of 1, bfloat16 precision, and an identical inference configuration. For each model, we use the 1,000 samples from MMAU test-mini, discard the first 5 samples as warm-up, and report the mean latency over the remaining 995 samples. CUDA synchronization is applied immediately before and after each timed operation.

KV Cache. We estimate the theoretical KV-cache footprint associated with the retained audio tokens as

$$M_{\text{KV}}^{\text{bytes}} = 2Ln\,(n_{\text{kv}} d_{\text{head}})\,b. \tag{15}$$

where L denotes the number of LLM layers, n the mean number of retained audio tokens, n_{kv} the number of key–value heads, d_{head} the head dimension, and b the number of bytes per element. The factor of 2 accounts for both key and value states. We report the KV-cache footprint in megabytes as $M_{\text{KV}}^{\text{MB}} = M_{\text{KV}}^{\text{bytes}}/1024^2$. For Qwen2.5-Omni-7B and AudioFlamingo-Next-Instruct, we use $L = 28$, $n_{\text{kv}} = 4$, $d_{\text{head}} = 128$, and $b = 2$ for bfloat16. For MOSS-Audio-8B-Instruct, we use $L = 36$, $n_{\text{kv}} = 8$, $d_{\text{head}} = 128$, and $b = 2$.

Performance. We report the downstream accuracy obtained under the same compression setting used for the efficiency measurements. Relative accuracy measures the retained performance with respect to Vanilla and is computed as

$$\text{Rel. Acc} = \frac{\text{Acc}_{\text{method}}}{\text{Acc}_{\text{Vanilla}}} \times 100. \tag{16}$$

F.2 Efficiency across Different ALLMs

To examine whether the efficiency benefits of TimePrune extend across different ALLMs, we further evaluate AudioFlamingo-Next-Instruct and MOSS-Audio-8B-Instruct on MMAU test-mini. These models complement the Qwen2.5-Omni-7B analysis in Sec. 4.4 by covering different model architectures and audio representation pipelines. Following the same evaluation protocol, we compare computational cost, prefilling latency, KV-cache footprint, and downstream performance under audio token compression.

Efficiency on AudioFlamingo-Next-Instruct. As shown in Table 18, TimePrune retains 99.2% of the Vanilla score at only 15% token retention, matching HeadRouter at 30% retention. Despite the additional compression cost, TimePrune reduces LLM FLOPs from 2.36T to 0.35T and the KV-cache footprint from 19.6 MB to 2.9 MB, while achieving a $1.83\times$ total prefilling speedup. These results show that the computational savings of TimePrune extend to AudioFlamingo-Next-Instruct while largely preserving its downstream score.

Efficiency on MOSS-Audio. As shown in Table 19, TimePrune retains 98.5% of the Vanilla score at only 15% token retention, while achieving a score comparable to HeadRouter at 30% retention. TimePrune reduces audio-token-dependent LLM FLOPs from 1.26T to 0.18T and the KV-cache

Table 18: **Inference efficiency on AudioFlamingo-Next-Instruct using MMAU test-mini.** FLOPs denote the theoretical audio-token-dependent LLM computation. Total includes compression overhead, and Rel. Score is measured relative to Vanilla.

Method	Retention Ratio R	FLOPs (T)	Prefilling Time			KV Cache (MB)	Score	Rel. Score (%)
			Compression	LLM Forward	Total			
Vanilla	100%	2.36	–	54.1	54.2 (1.0×)	19.6	0.756	100.0
VisionZip	30%	0.71	3.6	29.8	33.5 (1.62×)	5.9	0.738	97.6
DivPrune	30%	0.70	2.6	29.8	32.4 (1.67×)	5.9	0.735	97.2
Seg. Top-K	30%	0.67	0.8	29.5	30.3 (1.79×)	5.6	0.747	98.8
HeadRouter	30%	0.70	0.7	29.8	30.6 (1.77×)	5.9	0.750	**99.2**
TimePrune	**15%**	**0.35**	3.6	26.0	29.6 (**1.83**×)	**2.9**	0.750	**99.2**

footprint from 25.2 MB to 3.7 MB. Including the compression overhead, it reduces total prefilling time from 39.6 ms to 26.3 ms, corresponding to a $1.50\times$ speedup. These results further demonstrate that the efficiency gains of TimePrune extend to MOSS-Audio despite its different audio representation pipeline.

Table 19: **Inference efficiency on MOSS-Audio-8B-Instruct using MMAU test-mini.** FLOPs denote the theoretical audio-token-dependent LLM computation. Total includes compression overhead, and Rel. Score is measured relative to Vanilla.

Method	Retention Ratio R	FLOPs (T)	Prefilling Time			KV Cache (MB)	Score	Rel. Score (%)
			Compression	LLM Forward	Total			
Vanilla	100%	1.26	–	39.5	39.6 (1.0×)	25.2	0.747	100.0
VisionZip	30%	0.38	2.2	27.4	29.6 (1.33×)	7.6	0.721	96.5
DivPrune	30%	0.38	1.3	27.4	28.8 (1.38×)	7.6	0.716	95.9
Seg. Top-K	30%	0.33	0.5	26.7	27.2 (1.45×)	6.8	0.730	97.7
HeadRouter	30%	0.37	0.6	27.3	27.9 (1.42×)	7.5	0.734	98.3
TimePrune	**15%**	**0.18**	1.4	24.8	26.3 (**1.50**×)	**3.7**	0.736	**98.5**