HYPTOC: HYPERBOLIC VISION TOKEN COMPRESSION FOR VIDEO LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Video large language models (Video-LLMs) achieve strong video understanding capabilities, while the large number of vision tokens substantially increases inference cost. Recent training-free compression methods have attained promising efficiency by modeling token importance and redundancy in video representations. In this study, we investigate the geometry of video token representations and find that their embedding space exhibits pronounced tree-like organization compatible with hyperbolic geometry. Within individual frames, deeper vision features reveal increasing spatial hierarchy, whereas between frames, tokens corresponding to the same semantic object maintain a coherent temporal hierarchy despite substantial spatial displacement. Motivated by these findings, we introduce HypToC, a training-free framework that exploits hyperbolic geometry for video token compression with two new modules. Intra-Frame Lorentz Selection retains a fixed-size set of informative and geometrically diverse tokens. Inter-Frame Lorentz Clustering then aggregates the remaining tokens according to continuous lowest common ancestor relations in hyperbolic space. HypToC is training-free, plug-and-play, and model-agnostic. Experiments across representative Video-LLMs and benchmarks show that HypToC achieves state-of-the-art average performance under diverse matched token budgets.

1 INTRODUCTION

Video large language models (Video-LLMs) have recently shown strong capabilities in video understanding and temporal reasoning (Li et al., 2024a; Zhang et al., 2024b; Bai et al., 2025a). A video is usually encoded as a long sequence of vision tokens, which dominate the input length to a language model (Chen et al., 2026). As Transformer computation grows quadratically with sequence length (Vaswani et al., 2017), increasing the number of frames or spatial patches can lead to prohibitive memory and latency. Therefore, vision token compression has emerged as an important direction for efficient Video-LLMs, particularly in training-free settings preserving the vision encoder, projector, and language model (Chen et al., 2024; Shen et al., 2025; Fan et al., 2026).

Existing token compression methods typically select tokens by attention scores or merge redundant tokens by feature similarity (Chen et al., 2024; Yang et al., 2025b; Shen et al., 2025; Fan et al., 2026; Bolya et al., 2022). Although these strategies can reduce local redundancy, they largely overlook the hierarchical organization of video tokens. Within each frame, vision encoders progressively transform low-level patches into semantic structures by spatial and part-whole relations. Across frames, tokens associated with the same object or event remain semantically related despite motion, occlusion, and spatial misalignment. Video-token redundancy is therefore not merely pairwise but also hierarchical. A suitable compression method should therefore preserve the spatial hierarchy within each frame while capturing the temporal hierarchy across frames.

Hyperbolic space is naturally suited to hierarchical and tree-like data due to its negative curvature (Nickel & Kiela, 2017; Mettes et al., 2024). To determine whether video token representations exhibit such geometry, we quantify their tree-likeness using relative Gromov hyperbolicity, $\delta_{\mathrm{rel}} = 2\delta/\operatorname{diam}(X)$, following the standard four-point protocol with Euclidean distance (Gromov, 1987; Yang et al., 2025a). Fig. 1 reports the detailed analysis on Qwen3-VL, while Appendix A.1 further verifies the consistent hyperbolicity reduction across four backbones. As shown in Fig. 1(a), joint spatiotemporal representations attain lower relative Gromov hyperbolicity than per-frame, image, text, random, and spherical features, indicating a stronger affinity to tree-like geometry. This

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.

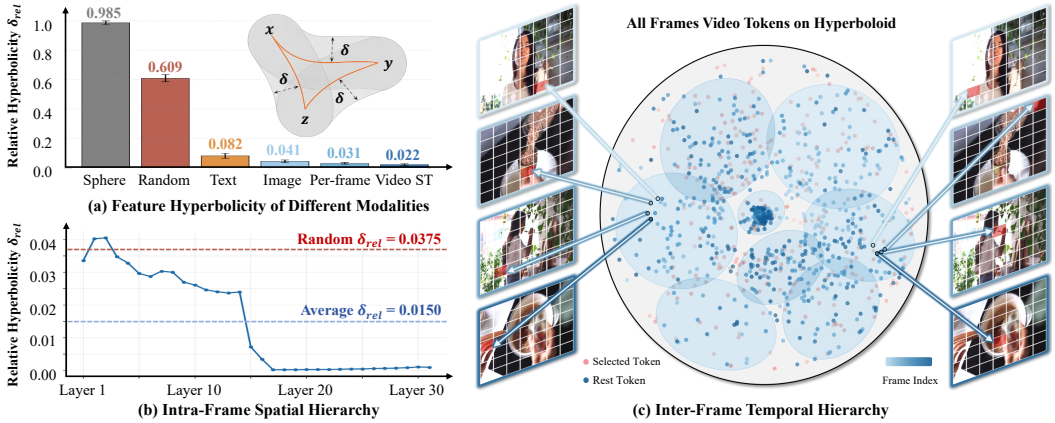


Figure 1: **Video tokens exhibit intra-frame spatial hierarchy and inter-frame temporal hierarchy.** (a) Relative Gromov hyperbolicity δ_{rel} across different feature sets under matched sample size, feature dimension, and distance metric. Video spatiotemporal features achieve the lowest δ_{rel}, indicating the strongest affinity to tree-like geometry. (b) Within each frame, relative hyperbolicity progressively decreases across deeper encoder layers. (c) Tokens associated with the same object may appear at different locations across frames but form coherent clusters after mapping to Lorentz space. Red and blue points denote selected and unselected tokens, respectively.

suggests that the hierarchy of video tokens becomes more pronounced through joint spatiotemporal organization rather than being inherited from image features alone. This organization appears at two complementary levels. At the intra-frame level, deeper vision features exhibit increasingly pronounced spatial hierarchy, as illustrated in Fig. 1(b). At the inter-frame level, tokens corresponding to the same semantic object form coherent groups after projection onto the Lorentz hyperboloid, even when they occupy different image locations, as shown in Fig. 1(c). Here, temporal hierarchy refers to hierarchical consistency among semantic observations distributed across frames.

Inspired by these observations, we propose HypToC, a training-free framework that exploits hyperbolic geometry for video token compression. Hyperbolic space admits several isometric models, among which we adopt the Lorentz model for its numerically stable distance and geodesic computations (Mishne et al., 2022). To preserve intra-frame structure, *Intra-Frame Lorentz Selection* formulates fixed-budget token selection as a quality-aware k-determinantal point process constructed from a Lorentz-distance kernel (Kulesza & Taskar, 2011). It retains informative tokens while maintaining geometric coverage within each frame. To reduce inter-frame redundancy, *Inter-Frame Lorentz Clustering* groups the residual tokens according to the depth of their continuous lowest common ancestors in hyperbolic space (Chami et al., 2020). It therefore aggregates semantically related tokens across frames without relying on rigid spatial correspondence. HypToC operates directly on projected vision tokens and does not require a hyperbolic backbone, retraining, fine-tuning, or architecture-specific modification. Our contributions can be summarized as follows.

- We reveal that video token representations exhibit a hierarchical structure, with joint spatiotemporal features being more tree-like than single-frame or single-modality features and showing distinct intra-frame spatial and inter-frame temporal hierarchies.
- We propose HypToC, a training-free hyperbolic framework that combines *Intra-Frame Lorentz Selection* for geometry-aware representative selection with *Inter-Frame Lorentz Clustering* for cross-frame hierarchy aggregation.
- Extensive experiments across multiple Video-LLMs and benchmarks show that HypToC achieves a superior accuracy–efficiency trade-off over strong recent training-free baselines while remaining plug-and-play and model-agnostic.

2 Related Work

Vision-Language Models and Video-LLMs. Vision-language models couple pretrained vision encoders with large language models, including Flamingo (Alayrac et al., 2022), BLIP-2 (Li et al., 2023), InstructBLIP (Dai et al., 2023), LLaVA (Liu et al., 2023), LLaVA-OneVision (Li et al., 2024a), and Qwen-VL (Bai et al., 2025a). Video-ChatGPT (Maaz et al., 2024), Video-

LLaMA (Zhang et al., 2023), and Video-LLaVA (Lin et al., 2024) extend this paradigm to video, while LLaMA-VID (Li et al., 2024c), MovieChat (Song et al., 2024), VideoStreaming (Qian et al., 2024), and LLaVA-Video (Zhang et al., 2024b) support long-form or streaming understanding. Recent models further scale temporal modeling and context capacity (Bai et al., 2025a; Zhang et al., 2025a; Ryoo et al., 2024), but dense frame sampling remains costly.

Vision Token Compression and Pruning for VLMs. Vision token compression reduces inference cost through pruning, merging, or compact projection, as explored by Token Merging (Bolya et al., 2022), FastV (Chen et al., 2024), LLaVA-PruMerge (Shang et al., 2025), TokenPacker (Li et al., 2025), VisionZip (Yang et al., 2025b), SparseVLM (Zhang et al., 2024a), PyramidDrop (Xing et al., 2024), and MustDrop (Liu et al., 2024). Other methods use visual cues, context, or adaptive reduction (Zhang et al., 2025c; Zhong et al., 2025; Endo et al., 2025; Wen et al., 2025). For video, LLaMA-VID (Li et al., 2024c), MovieChat (Song et al., 2024), PruneVID (Huang et al., 2025), FastVID (Shen et al., 2025), HoliTom (Shao et al., 2025), and FlashVID (Fan et al., 2026) model spatiotemporal redundancy or compact memory, while recent work considers trajectory curvature (Lin et al., 2026), local-global context (Li et al., 2026), and temporal fingerprints (Liu et al., 2026). HypToC exploits the tree-like organization of video tokens.

Hyperbolic Geometry and Hierarchical Representations. Hyperbolic space compactly models hierarchical and tree-like data through Poincaré embeddings (Nickel & Kiela, 2017) and Lorentz embeddings (Nickel & Kiela, 2018). Hyperbolic neural networks extend learning to negatively curved manifolds (Ganea et al., 2018; Chami et al., 2019) and visual tasks (Khrulkov et al., 2020; Mettes et al., 2024; Wang et al., 2025). Hyperbolic hierarchical clustering connects continuous embeddings with discrete trees through continuous lowest common ancestors (Chami et al., 2020). Unlike prior work that learns hyperbolic representations or known hierarchies from training data, HypToC exploits the tree-like organization of pretrained video tokens for training-free compression.

3 PRELIMINARIES

Hyperbolic geometry. Hyperbolic space is a Riemannian manifold with constant negative curvature $\kappa < 0$ and admits several isometric models, including the Klein, Poincaré ball, and Lorentz models (Cannon et al., 1997). In this work, we adopt the Lorentz model for its numerical stability in hyperbolic learning (Mishne et al., 2022). The m-dimensional Lorentz model is defined as

$$\mathbb{H}^m_\kappa = \left\{ \mathbf{z} \in \mathbb{R}^{m+1} \mid \langle \mathbf{z}, \mathbf{z} \rangle_{\mathcal{L}} = \frac{1}{\kappa},\ z_0 > 0 \right\}, \quad \langle \mathbf{u}, \mathbf{v} \rangle_{\mathcal{L}} = -u_0 v_0 + \sum_{r=1}^{m} u_r v_r, \tag{1}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{L}}$ denotes the Lorentzian inner product. The geodesic distance between $\mathbf{z}, \mathbf{z}' \in \mathbb{H}^m_\kappa$ is given by

$$d_{\mathbb{H}_\kappa}(\mathbf{z}, \mathbf{z}') = \frac{1}{\sqrt{-\kappa}} \operatorname{arcosh}\left(\kappa \langle \mathbf{z}, \mathbf{z}' \rangle_{\mathcal{L}}\right). \tag{2}$$

The canonical origin of the Lorentz model is written as $\mathbf{o}_\kappa = [1/\sqrt{-\kappa}, 0, \ldots, 0]^\top$. For a point $\mathbf{z} \in \mathbb{H}^m_\kappa$, the tangent space $T_{\mathbf{z}}\mathbb{H}^m_\kappa$ provides a local linear approximation of the manifold. In the Lorentz model, it can be characterized as $T_{\mathbf{z}}\mathbb{H}^m_\kappa = \{\mathbf{v} \in \mathbb{R}^{m+1} \mid \langle \mathbf{v}, \mathbf{z} \rangle_{\mathcal{L}} = 0\}$. Given a tangent vector $\mathbf{v} \in T_{\mathbf{z}}\mathbb{H}^m_\kappa$, the exponential map at $\mathbf{z}$ is given by

$$\exp^\kappa_{\mathbf{z}}(\mathbf{v}) = \cosh(\alpha)\mathbf{z} + \frac{\sinh(\alpha)}{\alpha}\mathbf{v}, \quad \alpha = \sqrt{-\kappa}\|\mathbf{v}\|_{\mathcal{L}}. \tag{3}$$

Fixed-size determinantal point processes. Fixed-budget subset selection should retain informative items while avoiding redundant choices. Determinantal point processes (DPPs) model these two criteria jointly through a positive semidefinite kernel $\mathbf{L} \succeq 0$ indexed by a ground set $\mathcal{Y}$. For a subset $S \subseteq \mathcal{Y}$, let $\mathbf{L}_S$ denote the principal submatrix indexed by S. The determinant $\det(\mathbf{L}_S)$ measures the joint quality and diversity of the selected items: it increases when the items are informative and mutually distinct, and decreases when they are redundant. When exactly k items must be selected, conditioning a DPP on the constraint $|S| = k$ yields a fixed-size DPP, or k-DPP (Kulesza & Taskar, 2011). For any subset $S \subseteq \mathcal{Y}$ satisfying $|S| = k$, its probability is defined as

$$P^k_{\mathbf{L}}(S) = \frac{\det(\mathbf{L}_S)}{\sum_{S' \subseteq \mathcal{Y}, |S'|=k} \det(\mathbf{L}_{S'})}. \tag{4}$$

The budget fixes $|S|$, while the determinant scores the selected items jointly.

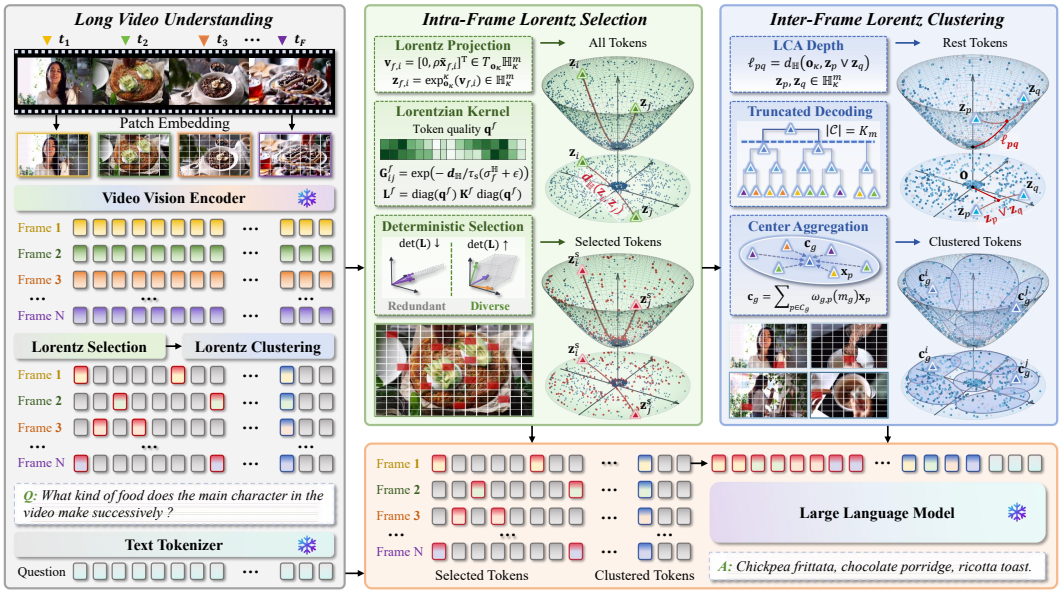


Figure 2: **Framework overview of the proposed HypToC.** (1) Intra-Frame Lorentz Selection maps frame-wise tokens to the Lorentz model, and performs quality-aware k-DPP selection to preserve informative and diverse tokens. (2) Inter-Frame Lorentz Clustering groups the remaining tokens by continuous lowest common ancestor (LCA) depth in the Lorentz model, and aggregates each cluster as a representative center token (i.e., a merged token). The selected and merged tokens are then fed into the frozen LLM for efficient long video understanding.

4 METHOD

4.1 OVERVIEW

HypToC is a training-free video token compression framework inserted between the vision encoder and the language model. Given F sampled frames, the vision encoder produces the token set $\mathbf{X} = \{\mathbf{x}_{f,i} \in \mathbb{R}^d \mid f \in [1, F], i \in [1, N]\}$, where N denotes the number of tokens per frame and d denotes their feature dimension. The full sequence contains $M = FN$ vision tokens, and a retention ratio $R \in (0, 1]$ determines the compressed budget $K = \lfloor RM \rfloor$.

HypToC allocates this budget between two complementary operations. *Intra-Frame Lorentz Selection* retains informative and geometrically diverse anchor tokens from each frame, whereas *Inter-Frame Lorentz Clustering* aggregates the residual tokens according to their continuous lowest common ancestor relations across frames. The anchor ratio $\alpha \in [0, 1]$ splits the budget into

$$k_s = \left\lfloor \frac{\alpha K}{F} \right\rfloor, \quad K_s = Fk_s, \quad K_m = K - K_s. \tag{5}$$

Here, k_s tokens are retained from each frame, resulting in K_s anchor tokens and K_m merged tokens. After both stages, the anchor and merged tokens are ordered according to their inherited frame and spatial positions, yielding a compressed sequence $\widehat{\mathbf{X}}$ of length $|\widehat{\mathbf{X}}| = K_s + K_m = K$.

Lorentz token embedding. Both compression stages operate on a Lorentz representation of the projected vision tokens. Following the standard Lorentz projection mapping (Ganea et al., 2018; Yang et al., 2025a), we scale each token by a factor $\rho > 0$:

$$\mathbf{v}_{f,i} = \begin{bmatrix} 0 \\ \rho \mathbf{x}_{f,i} \end{bmatrix} \in T_{\mathbf{o}_\kappa} \mathbb{H}^m_\kappa, \qquad \mathbf{z}_{f,i} = \exp^\kappa_{\mathbf{o}_\kappa}(\mathbf{v}_{f,i}) \in \mathbb{H}^m_\kappa. \tag{6}$$

Here, $\kappa < 0$ denotes the sectional curvature and ρ is a scaling factor. We use $\kappa = -1.0$ and $\rho = 0.02$ unless otherwise specified. Parameter sensitivity is reported in Appendix B.6.

4.2 INTRA-FRAME LORENTZ SELECTION

Within frame f, anchor selection is a fixed-size subset problem over $\mathcal{Y}_f = [N]$. We select k_s tokens using both token quality and subset-level redundancy, rather than ranking tokens by attention alone.

Token quality. The quality score combines attention saliency with cross-frame centrality. Let $a_{f,i}$ denote the attention assigned to token i by the `[CLS]` token when it is available. Otherwise, $a_{f,i}$ is computed by averaging the attention assigned to token i by all tokens within the frame. To measure cross-frame centrality, each frame g is summarized by the mean of its normalized token features. The frame summary $\mathbf{h}_g$, centrality score $r_{f,i}$, and token quality $q_{f,i}$ are computed as

$$\mathbf{h}_g = \frac{1}{N}\sum_{j=1}^{N}\frac{\mathbf{x}_{g,j}}{\|\mathbf{x}_{g,j}\|_2+\epsilon}, \quad r_{f,i} = \frac{1}{F}\sum_{g=1}^{F}\cos\left(\mathbf{x}_{f,i},\mathbf{h}_g\right), \quad q_{f,i} = \left(\bar{a}_{f,i}+\epsilon\right)\left(\bar{r}_{f,i}+\epsilon\right), \tag{7}$$

where $\bar{a}_{f,i}$ and $\bar{r}_{f,i}$ denote min–max normalization over the tokens in frame f. For subsequent kernel construction, the quality scores of frame f are collected into $\mathbf{q}^f = [q_{f,1},\ldots,q_{f,N}]^\top$.

Lorentzian determinantal kernel. Token quality alone does not account for redundancy among the selected tokens. We therefore construct a k-DPP kernel from pairwise Lorentz distances within each frame. To normalize the distance scale across frames and model families, we define

$$\begin{aligned}\sigma_f^{\mathbb{H}} &= \mathrm{median}_{1\le i<j\le N}\, d_{\mathbb{H}_\kappa}\left(\mathbf{z}_{f,i},\mathbf{z}_{f,j}\right),\\ G_{ij}^f &= \exp\left(-\frac{d_{\mathbb{H}_\kappa}\left(\mathbf{z}_{f,i},\mathbf{z}_{f,j}\right)}{\tau_s\left(\sigma_f^{\mathbb{H}}+\epsilon\right)}\right), \qquad \mathbf{K}^f = \mathbf{G}^f + \epsilon\mathbf{I},\end{aligned} \tag{8}$$

where $\tau_s > 0$ is a dimensionless selection temperature. The positive-semidefiniteness of $\mathbf{K}^f$ is established in Appendix A.2. The quality-aware Lorentzian k-DPP kernel and its subset determinant are

$$\begin{aligned}\mathbf{L}^f &= \mathrm{diag}(\mathbf{q}^f)\mathbf{K}^f\,\mathrm{diag}(\mathbf{q}^f),\\ \det(\mathbf{L}_S^f) &= \left(\prod_{i\in S} q_{f,i}^2\right)\det(\mathbf{K}_S^f), \qquad S\subseteq[N].\end{aligned} \tag{9}$$

Deterministic fixed-budget selection. HypToC seeks its maximum a posteriori (MAP) subset, rather than sampling from the k-DPP. Let $\mathcal{S}_{k_s} = \{S\subseteq[N] \mid |S| = k_s\}$. The optimal subset for frame f is defined as

$$S_f^{\mathrm{MAP}} = \arg\max_{S\in\mathcal{S}_{k_s}} \log\det\left(\mathbf{L}_S^f\right). \tag{10}$$

Because exact maximization is combinatorial, we approximate Eq. 10 using greedy determinant maximization. Starting from $S_f^{(0)} = \emptyset$, the selected token at iteration t is given by

$$i_t^\star = \arg\max_{i\in[N]\setminus S_f^{(t-1)}} \log\det\left(\mathbf{L}^f_{S_f^{(t-1)}\cup\{i\}}\right). \tag{11}$$

We then update $S_f^{(t)} = S_f^{(t-1)}\cup\{i_t^\star\}$ until $|S_f^{(t)}| = k_s$. The resulting anchor index set is defined as $\mathcal{A} = \{(f,i) \mid f\in[F],\, i\in S_f^{(k_s)}\}$, and the corresponding Euclidean vision tokens are retained unchanged in the compressed sequence. For efficient implementation, we maintain incremental Cholesky factors during greedy selection, avoiding repeated determinant computation from scratch.

4.3 INTER-FRAME LORENTZ CLUSTERING

After intra-frame selection, the residual token indices are collected as $\mathcal{U} = \mathcal{I}\setminus\mathcal{A}$, where $\mathcal{I} = [F]\times[N]$. For any index $p = (f_p, i_p)\in\mathcal{I}$, we use the shorthand $x_p = x_{f_p,i_p}$, $z_p = z_{f_p,i_p}$, and $q_p = q_{f_p,i_p}$. HypToC groups these residual tokens into K_m clusters according to their shared ancestry in the Lorentz model. The clustering affinity is order-agnostic and does not explicitly use frame distance or chronological order. Instead, it captures hierarchical consistency among semantic evidence distributed across frames.

Continuous LCA depth. In a rooted tree, the lowest common ancestor (LCA) of two nodes represents their deepest shared ancestor. Chami et al. (2020) extend this discrete notion to continuous embeddings in the Poincaré ball by treating the origin as the root. Since the Poincaré ball and Lorentz model are isometric representations of the same hyperbolic space, we treat the Lorentz origin $\mathbf{o}_\kappa$ as the root. Let $\mathrm{geo}(\mathbf{z}_p, \mathbf{z}_q)$ denote the geodesic segment connecting $\mathbf{z}_p, \mathbf{z}_q \in \mathbb{H}^m_\kappa$. Their continuous LCA is defined as the point on this geodesic closest to the origin:

$$\mathbf{z}_p \vee \mathbf{z}_q = \operatorname*{arg\,min}_{\mathbf{z} \in \mathrm{geo}(\mathbf{z}_p, \mathbf{z}_q)} d_{\mathbb{H}_\kappa}(\mathbf{o}_\kappa, \mathbf{z}). \tag{12}$$

Using the closed-form computation in Appendix A.3, we define the hierarchical affinity between tokens p and q as the depth of their continuous LCA, with larger values indicating deeper shared ancestry:

$$\ell_{pq} = d_{\mathbb{H}_\kappa}(\mathbf{o}_\kappa, \mathbf{z}_p \vee \mathbf{z}_q) = \frac{1}{\sqrt{-\kappa}} \operatorname{arcosh}\left(\sqrt{-\kappa}(\mathbf{z}_p \vee \mathbf{z}_q)_0\right). \tag{13}$$

Truncated LCA-depth clustering. To obtain exactly K_m clusters, we sort all residual-token pairs in decreasing order of ℓ_{pq}. Starting from the singleton partition $\mathcal{C}^{(0)} = \{\{p\} \mid p \in \mathcal{U}\}$, we process the sorted pairs sequentially and merge the clusters containing p and q whenever they are distinct. The procedure stops once $|\mathcal{C}| = K_m$, thereby matching the budget without decoding a full rooted tree. In practice, LCA depths are computed blockwise with a streaming pair buffer, avoiding materialization and global sorting of the full residual-token pair matrix.

Center-aware aggregation. For each cluster C_g, we choose the member with the largest total LCA depth to the other members as its center:

$$m_g = \operatorname*{arg\,max}_{p \in C_g} \sum_{\substack{q \in C_g \\ q \neq p}} \ell_{pq}. \tag{14}$$

For a singleton cluster, its only member serves as the center. As the language model consumes Euclidean vision embeddings, each cluster is aggregated in the original feature space. We combine token quality with proximity to the cluster center, such that informative and geometrically representative tokens receive larger weights. For scale-calibrated aggregation, we define $\sigma_m = \max\left\{\mathrm{median}_{g \in [K_m],\, p \in C_g,\, p \neq m_g} d_{\mathbb{H}_\kappa}\left(\mathbf{z}_p, \mathbf{z}_{m_g}\right), \epsilon\right\}$. Given a merging temperature $\tau_m > 0$, the aggregation score $s_{g,p}$, normalized weight $\omega_{g,p}$, and merged token $\mathbf{c}_g$ are computed as

$$s_{g,p} = \log(q_p + \epsilon) - \frac{d_{\mathbb{H}_\kappa}\left(\mathbf{z}_p, \mathbf{z}_{m_g}\right)}{\tau_m \sigma_m}, \qquad \omega_{g,p} = \frac{\exp(s_{g,p})}{\sum_{r \in C_g} \exp(s_{g,r})}, \qquad \mathbf{c}_g = \sum_{p \in C_g} \omega_{g,p} \mathbf{x}_p. \tag{15}$$

Final compressed token sequence. For the final sequence, $\pi(p) = (f_p, i_p)$ denotes the original token position, each merged token inherits $\pi(m_g)$ from its center, and $\mathrm{Sort}_{\mathrm{pos}}$ sorts by the attached positions. The final compressed vision token sequence is

$$\widehat{\mathbf{X}} = \mathrm{Sort}_{\mathrm{pos}}\left(\{(\mathbf{x}_p, \pi(p)) \mid p \in \mathcal{A}\} \cup \{(\mathbf{c}_g, \pi(m_g)) \mid g \in [K_m]\}\right). \tag{16}$$

Since $|\mathcal{A}| = K_s$ and $|\mathcal{C}| = K_m$, the resulting sequence satisfies $|\widehat{\mathbf{X}}| = K_s + K_m = K$. Anchor tokens retain their original positions, while each merged token inherits the position of its center. Sorting these inherited positions preserves the temporal order of the compressed sequence. Implementation details are provided in Appendix B.5.

5 Experiments

5.1 Experimental Setting

Evaluation Protocol. We evaluate HypToC on four video understanding benchmarks, namely, VideoMME (Fu et al., 2025), EgoSchema (Mangalam et al., 2023), LongVideoBench (Wu et al., 2024), and MVBench (Li et al., 2024b). These benchmarks cover short clips, egocentric videos, long-form videos, and temporally demanding multi-choice tasks. VideoMME covers 900 videos and 2,700 questions across short, medium, and long durations. EgoSchema targets long-range egocentric

Table 1: Comparison with training-free methods on Qwen2.5-VL and Qwen3-VL.

Method	Retention Ratio R	VideoMME				EgoSchema		LongVideo Bench	MVBench	Avg.	
		Short	Medium	Long	Overall	Subset	Total			Score	Rel. Acc (%)
Qwen3-VL											
Full Tokens	100%	74.9	60.0	56.4	63.8	65.4	68.3	58.0	65.3	63.9	100.0
AOT	25%	72.1	55.4	51.0	59.5	67.0	67.8	58.0	65.1	62.6	98.0
FlashVID	25%	73.0	56.6	53.8	61.1	66.4	67.0	59.1	64.7	63.0	98.6
HypToC	25%	**74.6**	**60.0**	**54.6**	**63.0**	**67.2**	**67.9**	**59.9**	**65.8**	**64.2**	**100.5**
AOT	20%	71.6	54.8	50.9	59.1	66.2	67.0	56.7	63.6	61.6	96.4
FlashVID	20%	72.6	56.4	53.6	60.9	64.4	66.1	58.0	63.9	62.2	97.5
HypToC	20%	**74.4**	**60.6**	**53.9**	**63.0**	**66.6**	**67.2**	**58.8**	**65.2**	**63.6**	**99.5**
AOT	15%	70.3	53.6	49.8	57.9	65.4	66.2	55.2	62.7	60.5	94.7
FlashVID	15%	70.1	55.1	52.9	59.4	63.0	65.3	57.6	62.4	61.2	95.8
HypToC	15%	**74.2**	**60.1**	**53.6**	**62.6**	**65.4**	**66.8**	**57.8**	**64.6**	**63.0**	**98.6**
AOT	10%	69.5	52.9	48.9	57.1	64.2	65.0	54.2	61.7	59.5	93.1
FlashVID	10%	69.2	54.4	52.0	58.6	61.8	64.6	56.6	61.5	60.3	94.5
HypToC	10%	**71.6**	**57.3**	**54.0**	**61.0**	**65.4**	**66.0**	**57.5**	**62.9**	**61.9**	**96.9**
Qwen2.5-VL											
Full Tokens	100%	72.6	61.4	49.9	61.3	60.2	58.3	58.9	68.0	61.6	100.0
FastVID	25%	69.9	56.3	49.7	58.6	57.2	56.4	57.8	64.7	59.4	96.4
AOT	25%	70.0	55.4	49.2	58.2	59.1	57.8	56.5	66.7	59.8	97.1
FlashVID	25%	70.4	58.6	49.7	59.6	**59.2**	56.8	58.1	66.5	60.2	97.7
HypToC	25%	**71.4**	**58.8**	**50.8**	**60.3**	58.8	**57.1**	**58.6**	**67.5**	**60.9**	**98.8**
FastVID	10%	66.3	53.6	49.0	56.3	56.0	55.6	55.4	62.3	57.4	93.2
AOT	10%	67.9	53.0	47.1	56.0	58.0	56.6	54.1	65.7	58.1	94.3
FlashVID	10%	68.1	54.7	49.0	57.3	57.4	55.9	57.1	65.5	58.9	95.6
HypToC	10%	**69.1**	**57.7**	**49.2**	**58.7**	**57.6**	**56.2**	**57.4**	**66.3**	**59.7**	**96.8**

reasoning from 250 hours of first-person video, while LongVideoBench evaluates long-context multimodal reasoning with 3,763 videos and 6,678 questions. MVBench contains 20 tasks for temporal understanding beyond static perception. Following the official protocols of each benchmark, we report multiple-choice accuracy for all evaluations, including the short, medium, long, and overall splits of VideoMME and the subset and full-set scores of EgoSchema.

Baselines. We compare HypToC with strong training-free token compression methods, including FastV (Chen et al., 2024), VisionZip (Yang et al., 2025b), FastVID (Shen et al., 2025), FlashVID (Fan et al., 2026), and AOT (Li et al., 2026). AOT is reimplemented on the Qwen models under the same evaluation protocol. All baselines are compared under matched token retention ratios. Baseline comparability is discussed in Appendix B.1.

Implementation. We conduct experiments on four representative Video-LLMs, including Qwen2.5-VL (Bai et al., 2025b), Qwen3-VL (Bai et al., 2025a), LLaVA-OneVision (Li et al., 2024a), and LLaVA-Video (Zhang et al., 2024b). For fair comparison, we follow the model-specific frame sampling, vision tokenization, and evaluation settings of FlashVID (Fan et al., 2026). For methods that compress tokens at different stages, we align the average vision-token budget during inference. All experiments are implemented with LMMs-Eval (Zhang et al., 2025b) and conducted on four NVIDIA H100 94G GPUs.

5.2 Token Compression Results

On Qwen Models. Table 1 reports results on Qwen2.5-VL and Qwen3-VL across multiple retention ratios. HypToC achieves the best average score at every evaluated budget on both Qwen model families. On Qwen3-VL, it outperforms AOT by 1.6–2.5 average points across $R = 25\%$–10%, while consistently outperforming FlashVID. At $R = 25\%$, HypToC reaches an average score of 64.2, exceeding FlashVID and AOT by 1.2 and 1.6 points, respectively. At $R = 15\%$, it improves the average score and VideoMME accuracy over FlashVID by 1.8 and 3.2 points. Even at $R = 10\%$, HypToC maintains 96.9% relative accuracy and exceeds FlashVID by 1.6 points in average score.

On LLaVA Models. Table 2 compares HypToC with baselines on LLaVA-OneVision and LLaVA-Video at 20% and 10% retention ratio. On LLaVA-OneVision, HypToC achieves the highest average

Table 2: Comparison with training-free methods on LLaVA-OneVision and LLaVA-Video.

Method	Retention Ratio R	VideoMME				EgoSchema		LongVideo Bench	MVBench	Avg.	
		Short	Medium	Long	Overall	Subset	Total			Score	Rel. Acc (%)
LLaVA-OneVision											
Full Tokens	100%	69.9	56.7	48.9	58.5	62.2	60.3	56.6	58.3	58.4	100.0
VisionZip	20%	68.6	57.0	48.3	58.0	62.0	60.0	55.4	57.6	57.7	98.8
FastVID	20%	69.9	56.3	47.4	57.9	61.2	59.5	55.9	58.1	57.9	99.1
FlashVID	20%	70.1	55.4	48.9	58.2	**63.0**	60.1	**58.5**	**58.2**	58.7	100.5
HypToC	20%	**71.8**	**57.1**	**49.6**	**59.5**	62.8	**60.1**	57.6	57.9	**58.8**	**100.6**
VisionZip	10%	60.3	52.9	46.7	53.3	61.6	58.5	49.4	54.8	54.0	92.5
FastVID	10%	68.1	55.7	47.8	57.2	58.8	58.7	55.7	57.0	57.1	97.8
FlashVID	10%	67.3	57.1	49.0	57.8	62.4	**60.0**	56.5	**57.4**	57.9	99.1
HypToC	10%	**69.7**	**56.6**	**49.9**	**58.7**	**64.4**	59.6	**56.9**	57.1	**58.1**	**99.4**
LLaVA-Video											
Full Tokens	100%	77.0	62.1	53.3	64.2	59.4	57.3	59.5	61.9	60.7	100.0
VisionZip	20%	72.3	59.6	**53.3**	61.7	59.0	56.4	58.0	59.8	59.0	97.2
FastVID	20%	74.6	60.8	52.3	**62.6**	57.0	55.0	57.1	60.2	58.7	96.7
FlashVID	20%	74.1	60.0	52.3	62.2	58.4	56.4	58.7	59.8	59.3	97.7
HypToC	20%	**74.9**	**61.8**	50.1	62.3	**59.2**	**57.3**	**59.4**	**60.3**	**59.8**	**98.6**
VisionZip	10%	69.4	57.9	51.2	59.5	54.4	53.9	54.5	58.5	56.6	93.2
FastVID	10%	71.8	57.3	50.2	59.8	54.8	52.4	56.9	59.3	57.1	94.1
FlashVID	10%	72.2	59.1	51.2	60.9	57.2	54.9	57.7	59.3	58.2	95.9
HypToC	10%	**73.7**	**61.2**	**51.2**	**62.0**	**58.2**	**55.6**	**57.9**	**60.2**	**58.9**	**97.1**

score at both budgets, preserving 100.6% and 99.4% relative accuracy, respectively. At $R = 20\%$, it reaches 59.5 on VideoMME, while at $R = 10\%$ it also outperforms FlashVID (56.9 vs. 56.5) on LongVideoBench. On LLaVA-Video, HypToC likewise obtains the best average score under both settings. At $R = 20\%$, it reaches an average score of 59.8 with the best results on EgoSchema, LongVideoBench, and MVBench. At $R = 10\%$, HypToC raises accuracy from 60.9 to 62.0 on VideoMME and outperforms the other methods on the remaining three benchmarks simultaneously. Video-level paired significance analysis is provided in Appendix B.4.

5.3 Qualitative Analysis

On Intra-Frame Token Selection. Fig. 3 compares the vision tokens selected by HypToC and FlashVID across video frames. White regions denote tokens selected only by FlashVID, pink regions denote tokens selected by both methods, and red regions denote tokens selected only by HypToC. While the two methods retain a substantial number of common tokens, HypToC additionally selects tokens from informative regions, including text, facial features, food, and action-related body parts.

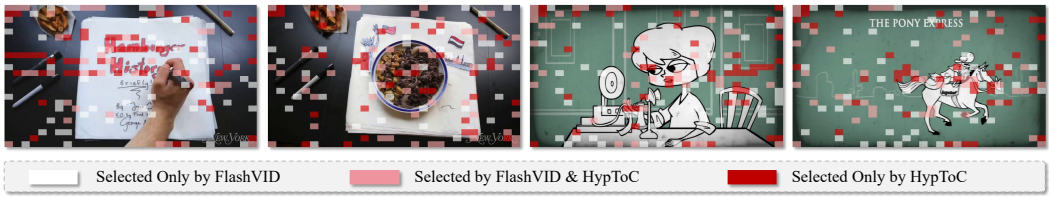


Figure 3: Visualization of intra-frame vision token selection on Qwen3-VL.

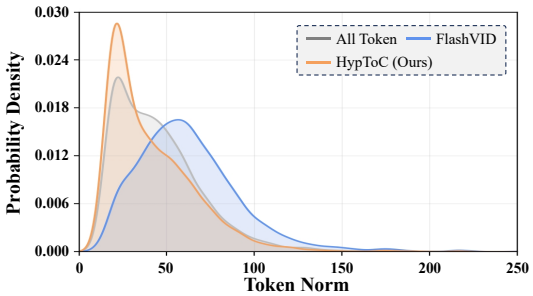


Figure 4: Token distributions on Qwen3-VL.

On Selected Token Norm Distribution. A statistical analysis of Qwen3-VL further examines the norm distribution of tokens selected by Intra-Frame Lorentz Selection. In Fig. 4, the x and y axes denote vision token norm and probability density, respectively, and all tokens are presented in gray, FlashVID tokens in blue, and HypToC tokens in orange. The HypToC distribution closely follows the dominant mode and overall shape of the full token set, whereas the FlashVID's shifts toward larger norms.

Table 3: Ablation study on Intra-Frame Lorentz Selection manifold on Qwen3-VL. ATS, ADTS denote attention-, attention-diversity (Fan et al., 2026), and E-DPP, L-DPP denote Euclidean and Lorentzian DPP token selection.

Method	VideoMME	EgoSchema	LongVideo Bench	MVBench	Rel. Acc.
ATS	56.2	63.5	55.1	58.4	91.3
ADTS	58.6	64.6	56.6	61.5	94.5
E-DPP	60.7	64.2	56.9	62.1	95.5
L-DPP	**61.0**	**66.0**	**57.5**	**62.9**	**96.9**

Table 4: Ablation study on α on Qwen3-VL. α controls the distribution of compression budget between Intra-Frame Lorentz Selection and Inter-Frame Lorentz Clustering.

α	VideoMME	EgoSchema	LongVideo Bench	MVBench	Rel. Acc.
0.0	58.7	64.8	55.9	61.4	94.3
0.4	60.2	65.5	56.8	**63.1**	96.2
0.6	60.7	65.9	57.5	62.5	96.6
0.8	**61.0**	66.0	**57.5**	62.9	**96.9**
1.0	59.8	**66.4**	57.1	61.8	96.0

Table 5: Comparison with training-free token compression methods on Qwen2.5-VL, Qwen3-VL and LLaVA-OneVision at ultra low retention ratio on VideoMME.

Method	Retention Ratio R	*Qwen2.5-VL*				*Qwen3-VL*				*LLaVA-OneVision*			
		Short	Medium	Long	Overall	Short	Medium	Long	Overall	Short	Medium	Long	Overall
Full Tokens	100%	72.6	61.4	49.9	61.3	74.9	60.0	56.4	63.8	69.9	56.7	48.9	58.5
FlashVID	2%	61.3	50.4	46.2	52.7	57.8	**50.3**	47.2	51.8	55.2	47.6	43.0	48.6
HypToC	2%	**62.0**	**51.4**	**46.7**	**53.4**	**57.9**	50.1	**49.4**	**52.5**	**55.8**	**49.7**	**44.1**	**49.9**
FlashVID	1%	54.1	49.0	45.6	49.6	**54.6**	48.8	46.6	50.0	46.4	45.1	43.7	45.1
HypToC	1%	**56.8**	**51.3**	**46.6**	**51.6**	53.4	**49.4**	**48.8**	**50.6**	**49.3**	**46.3**	**44.2**	**46.6**
FlashVID	0.5%	51.7	46.7	44.0	47.4	50.8	47.1	45.6	47.8	44.9	43.1	42.9	43.7
HypToC	0.5%	**53.4**	**47.0**	**44.7**	**48.4**	**51.1**	**47.3**	**45.8**	**48.1**	**45.1**	**44.6**	**43.2**	**44.2**

5.4 Ablation Studies

On Selection Geometry. Table 3 progressively isolates the effects of token importance, diversity modeling, determinantal selection, and Lorentz geometry. ATS performs attention-only selection, while ADTS additionally models token diversity. E-DPP and L-DPP share the same token quality, input features, fixed budget, and MAP solver, differing only in their Euclidean and Lorentz affinity kernels. ADTS improves relative accuracy from 91.3% to 94.5%, and E-DPP further reaches 95.5%. L-DPP achieves the best relative accuracy of 96.9%.

On Selection and Clustering Balance. Table 4 analyzes the budget ratio α between Intra-Frame Lorentz Selection and Inter-Frame Lorentz Clustering. When $\alpha = 0$, all compressed tokens are produced by clustering, yielding 94.3% relative accuracy. When $\alpha = 1$, the budget is dominated by anchor selection, yielding 96.0% relative accuracy. The best performance occurs at $\alpha = 0.8$, achieving 96.9% relative accuracy and the highest scores on VideoMME, LongVideoBench, and MVBench. This result exceeds both endpoint settings, indicating that jointly allocating the token budget to selection and clustering is beneficial. The close performance at $\alpha = 0.6$ further suggests that HypToC remains stable when the budget is reasonably shared between the two modules. Further ablations of clustering affinity, aggregation strategy, and the transferability of α are provided in Appendix B.7.

On Ultra-Low Retention Ratios. Table 5 evaluates HypToC on VideoMME under extremely small token budgets. At $R = 2\%$, HypToC outperforms FlashVID on overall accuracy by 0.7 points on Qwen2.5-VL, 0.7 points on Qwen3-VL, and 1.3 points on LLaVA-OneVision. At $R = 1\%$, the gains become 2.0, 0.6, and 1.5 points. Even at the challenging 0.5% retention ratio, HypToC still outperforms FlashVID on all three model families. When only a tiny fraction of vision tokens can be retained, attention-only saliency or flat redundancy estimation becomes less reliable. As a result, HypToC remains robust even when the token budget is reduced to below 2%.

6 Conclusion

We proposed HypToC, a training-free hyperbolic framework for vision token compression in Video-LLMs. Guided by intra-frame spatial hierarchy and inter-frame temporal hierarchy, HypToC introduces Intra-Frame Lorentz Selection for geometry-aware anchor selection and Inter-Frame Lorentz Clustering for hierarchy-aware aggregation. Experiments across multiple Video-LLMs and benchmarks show state-of-the-art average accuracy under aggressive token budgets, demonstrating the value of exploiting the tree-like organization of pretrained video tokens. Future directions include exploring the hierarchical potential of video representations through hyperbolic training.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

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*, 2025b.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. *arXiv preprint arXiv:2210.09461*, 2022.

James W. Cannon, William J. Floyd, Richard Kenyon, and Walter R. Parry. Hyperbolic geometry. In *Flavors of Geometry*, volume 31 of *MSRI Publications*, pp. 59–115. Cambridge University Press, 1997.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in Neural Information Processing Systems*, 32:4868–4879, 2019.

Ines Chami, Albert Gu, Vaggos Chatziafratis, and Christopher Ré. From trees to continuous embeddings and back: Hyperbolic hierarchical clustering. *Advances in Neural Information Processing Systems*, 33:15065–15076, 2020.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision*, pp. 19–35. Springer, 2024.

Xueyi Chen, Keda Tao, Kele Shao, and Huan Wang. StreamingTOM: Streaming token compression for efficient video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24675–24685, 2026.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36:49250–49267, 2023.

Mark Endo, Xiaohan Wang, and Serena Yeung-Levy. Feather the throttle: Revisiting visual token pruning for vision-language model acceleration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22826–22835, 2025.

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. *arXiv preprint arXiv:2602.08024*, 2026.

Aasa Feragen, Francois Lauze, and Soren Hauberg. Geodesic exponential kernels: When curvature and linearity conflict. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3032–3042, 2015.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24108–24118, 2025.

Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. *Advances in Neural Information Processing Systems*, 31:5345–5355, 2018.

Mikhael Gromov. Hyperbolic groups. In *Essays in Group Theory*, pp. 75–263. Springer, 1987.

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

Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6418–6428, 2020.

Alex Kulesza and Ben Taskar. k-DPPs: Fixed-size determinantal point processes. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 1193–1200, 2011.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Jinlong Li, Liyuan Jiang, Haonan Zhang, and Nicu Sebe. Token reduction via local and global contexts optimization for efficient video large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10451–10461, 2026.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, pp. 19730–19742. PMLR, 2023.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. MVBench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22195–22206, 2024b.

Wentong Li, Yuqian Yuan, Jian Liu, Dongqi Tang, Song Wang, Jie Qin, Jianke Zhu, and Lei Zhang. TokenPacker: Efficient visual projector for multimodal LLM. *International Journal of Computer Vision*, 133(10):6794–6812, 2025.

Yanwei Li, Chengyao Wang, and Jiaya Jia. LLaMA-VID: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pp. 323–340. Springer, 2024c.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-LLaVA: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5971–5984, 2024.

Xinying Lin, Xuyang Liu, Yiyu Wang, Teng Ma, and Wenqi Ren. V-CAST: Video curvature-aware spatio-temporal pruning for efficient video large language models. *arXiv preprint arXiv:2603.27650*, 2026.

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

Ting Liu, Liangtao Shi, Richang Hong, Yue Hu, Quanjun Yin, and Linfeng Zhang. Multi-stage vision token dropping: Towards efficient multimodal large language model. *arXiv preprint arXiv:2411.10803*, 2024.

Xinxin Liu, Shiwei Gan, Xiao Liu, Yafeng Yin, Lei Xie, and Sanglu Lu. InfoMerge: Information-aware token compression for efficient video large language models. *arXiv preprint arXiv:2606.02161*, 2026.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 12585–12602, 2024.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. EgoSchema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36:46212–46244, 2023.

Pascal Mettes, Mina Ghadimi Atigh, Martin Keller-Ressel, Jeffrey Gu, and Serena Yeung. Hyperbolic deep learning in computer vision: A survey. *International Journal of Computer Vision*, 132 (9):3484–3508, 2024.

Gal Mishne, Zhengchao Wan, Yusu Wang, and Sheng Yang. The numerical stability of hyperbolic representation learning. *arXiv preprint arXiv:2211.00181*, 2022.

Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in Neural Information Processing Systems*, 30:6338–6347, 2017.

Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*, pp. 3779–3788. PMLR, 2018.

Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. *Advances in Neural Information Processing Systems*, 37:119336–119360, 2024.

Guyan Robertson. Crofton formulae and geodesic distance in hyperbolic spaces. *arXiv preprint arXiv:1302.5790*, 2013.

Michael S Ryoo, Honglu Zhou, Shrikant Kendre, Can Qin, Le Xue, Manli Shu, Jongwoo Park, Kanchana Ranasinghe, Silvio Savarese, Ran Xu, et al. xGen-MM-Vid (BLIP-3-Video): You Only Need 32 Tokens to Represent a Video Even in VLMs. *arXiv preprint arXiv:2410.16267*, 2024.

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

Kele Shao, Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. HoliTOM: Holistic token merging for fast video large language models. *arXiv preprint arXiv:2505.21334*, 2025.

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

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. MovieChat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18221–18232, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30:5998–6008, 2017.

Ziwei Wang, Sameera Ramasinghe, Chenchen Xu, Julien Monteil, Loris Bazzani, and Thalaiyasingam Ajanthan. Learning visual hierarchies in hyperbolic space for image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9924–9934, 2025.

Zichen Wen, Yifeng Gao, Weijia Li, Conghui He, and Linfeng Zhang. Token pruning in multimodal large language models: Are we solving the right problem? In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 15537–15549, 2025.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2024.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. PyramidDrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*, 2024.

Menglin Yang, Aosong Feng, Bo Xiong, Jiahong Liu, Irwin King, Rex Ying, et al. Hyperbolic fine-tuning for large language models. *Advances in Neural Information Processing Systems*, 38: 29613–29648, 2025a.

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*, pp. 19792–19802, 2025b.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. VideoLLaMA 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025a.

Hang Zhang, Xin Li, and Lidong Bing. Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 543–553, 2023.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. LMMs-Eval: Reality check on the evaluation of large multimodal models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 881–916, 2025b.

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*, pp. 20857–20867, 2025c.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. SparseVLM: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*, 2024a.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. LLaVA-Video: Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024b.

Yiwu Zhong, Zhuoming Liu, Yin Li, and Liwei Wang. AIM: Adaptive inference of multi-modal LLMs via token merging and pruning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20180–20192, 2025.

Supplementary Material

CONTENTS

A Geometry and Theoretical Analysis **15**

A.1 Hyperbolicity Analysis Protocol 15

A.2 Lorentz Projection Mapping 16

A.3 LCA Computation 16

B More Experimental Results **17**

B.1 Baseline Comparability 17

B.2 Additional Results on Qwen Models 18

B.3 Additional Results on LLaVA Models 18

B.4 Significance Analysis 18

B.5 Computation Efficiency 19

B.6 Lorentz Geometry Sensitivity 22

B.7 Ablation Analysis of Inter-Frame Lorentz Clustering 22

C Additional Qualitative Analysis **24**

C.1 Additional Intra-Frame Token Selection Results 24

C.2 Additional Vision Token Norm Distribution Results 24

D Case Studies **24**

D.1 Case Studies on Qwen3-VL 24

D.2 Case Studies on LLaVA-Video 25

D.3 Failure Case Analysis 25

E Usage of Large Language Models **26**

A Geometry and Theoretical Analysis

A.1 Hyperbolicity Analysis Protocol

We follow established embedding-space hyperbolicity analysis (Gromov, 1987; Yang et al., 2025a) and use relative Gromov hyperbolicity to quantify the tree-likeness of video token representations. For a metric space X, Gromov hyperbolicity is estimated by the four-point condition. Given four points $a, b, c, d \in X$, let $s_{(1)} \geq s_{(2)} \geq s_{(3)}$ denote the sorted values of

$$d(a,b) + d(c,d), \quad d(a,c) + d(b,d), \quad d(a,d) + d(b,c). \tag{17}$$

The four-point hyperbolicity is

$$\delta_4(a,b,c,d) = \frac{1}{2}\left(s_{(1)} - s_{(2)}\right). \tag{18}$$

Following prior embedding-space analyses (Yang et al., 2025a), we use Euclidean distance and normalize by the feature-space diameter:

$$\delta_{\mathrm{rel}}(X) = \frac{2\delta(X)}{\mathrm{diam}(X)}, \qquad \mathrm{diam}(X) = \max_{x,y \in X} d(x,y), \tag{19}$$

where $\delta(X) = \max_{q \in \mathcal{Q}} \delta_4(q)$ and $\mathcal{Q}$ contains 10^5 uniformly sampled quadruples. Lower δ_{rel} indicates stronger affinity to a tree metric. An exact tree metric satisfies $\delta = 0$ and therefore $\delta_{\mathrm{rel}} = 0$, providing the natural lower reference of the measure.

Feature construction. The detailed hyperbolicity analyses in Fig. 1(a) and Fig. 1(b) are conducted on VideoMME using Qwen3-VL, with features extracted from the final vision-encoder layer unless otherwise specified. For the Per-frame condition, relative hyperbolicity is computed independently from the $N = 196$ tokens of each sampled frame and then averaged across frames and videos. For the Video ST condition, each trial samples the same number of tokens uniformly across 20 frames of the same video. The Image, Text, Random, and Sphere controls in Fig. 1(a) use the same number of points per trial and the same feature dimension and distance metric.

To examine whether this trend generalizes across model families, we further repeat the Per-frame and Video ST analyses on Qwen2.5-VL, Qwen3-VL, LLaVA-OneVision, and LLaVA-Video, using matched token counts within each backbone and the same four-point protocol. Table 6 summarizes the results. Across all four backbones, Video ST representations consistently exhibit lower relative hyperbolicity than their Per-frame counterparts, with relative reductions of 25.0%–29.0%. This consistent trend indicates that the stronger tree-like organization of joint video representations is not specific to a single backbone.

Table 6: Cross-model relative Gromov hyperbolicity of Per-frame and Video ST representations. Values are reported as mean $\pm$ standard deviation over five repeated samplings. Lower values indicate stronger affinity to tree-like geometry.

Backbone	Per-frame	Video ST	Reduction
Qwen2.5-VL	0.034 ± 0.0016	0.025 ± 0.0011	26.5%
Qwen3-VL	0.031 ± 0.0013	0.022 ± 0.0008	29.0%
LLaVA-OneVision	0.036 ± 0.0022	0.027 ± 0.0015	25.0%
LLaVA-Video	0.035 ± 0.0019	0.026 ± 0.0012	25.7%

Sampling and aggregation. The analysis is performed over 100 videos with five repeated samplings, and we report the mean across all measurements. The same sampling protocol is used for all feature types. For the representative Qwen3-VL comparison, the Per-frame condition has a raw hyperbolicity of 1.25 ± 0.04 and a feature-space diameter of 80.6 ± 1.7, whereas Video ST has a raw hyperbolicity of 1.15 ± 0.03 and a diameter of 104.5 ± 2.3. Thus, Video ST exhibits both a smaller raw δ and a larger diameter, indicating that its lower relative hyperbolicity is not explained by diameter growth alone.

Layer-wise analysis. For Fig. 1(b), we apply the same relative-hyperbolicity protocol to token features extracted from different vision-encoder layers. All layers use the same videos, token samples, feature dimension, and distance metric. Because δ_{rel} is normalized by the representation diameter, the observed decrease cannot be attributed solely to uniform feature contraction.

A.2 Lorentz Projection Mapping

The hyperbolicity analyses in Fig. 1 are conducted on the original model representations before Lorentz mapping. Appendix A.1 further verifies the consistent hyperbolicity reduction across all four evaluated Video-LLM backbones. HypToC subsequently maps the vision tokens to the Lorentz model through Eq. 6. Specifically, we use

$$\mathbf{v}_{f,i} = \begin{bmatrix} 0 \\ \rho \mathbf{x}_{f,i} \end{bmatrix}, \qquad \mathbf{z}_{f,i} = \exp^{\kappa}_{\mathbf{o}_\kappa}(\mathbf{v}_{f,i}), \tag{20}$$

where $\rho > 0$ is a scaling factor. By radial isometry, the mapped token satisfies $d_{\mathbb{H}_\kappa}(\mathbf{o}_\kappa, \mathbf{z}_{f,i}) = \rho \|\mathbf{x}_{f,i}\|_2$. Thus, the mapping preserves token magnitude as radial information, while ρ controls its overall scale. Additional controls that replace token magnitudes with a constant norm, shuffle token norms, or shuffle token directions show that the performance gain cannot be explained by either magnitude or direction alone. The results are reported in Appendix B.6.

Following the curvature parameterization of HypLoRA (Yang et al., 2025a), we write $\kappa = -1/K$. We use $K = 1$, corresponding to $\kappa = -1$, and set $\rho = 0.02$ unless otherwise specified. Sensitivity to both parameters is reported in Appendix B.6.

Positive-definiteness of the Lorentz similarity. The geodesic distance on real hyperbolic space is conditionally negative definite (Robertson, 2013; Feragen et al., 2015). Positive rescaling preserves this property; therefore, for each frame,

$$\widehat{d}_f(\mathbf{z}_{f,i}, \mathbf{z}_{f,j}) = \frac{d_{\mathbb{H}_\kappa}(\mathbf{z}_{f,i}, \mathbf{z}_{f,j})}{\tau_s \left(\sigma_f^{\mathbb{H}} + \epsilon\right)} \tag{21}$$

is also conditionally negative definite. By Schoenberg's theorem, $\exp(-\widehat{d}_f)$ is positive semidefinite. Consequently, $\mathbf{K}^f = \mathbf{G}^f + \epsilon \mathbf{I} \succeq 0$ and

$$\mathbf{L}^f = \mathrm{diag}(\mathbf{q}^f) \mathbf{K}^f \mathrm{diag}(\mathbf{q}^f) \succeq 0. \tag{22}$$

Thus, no spectral projection is required. In implementation, pairwise distances are computed symmetrically and a small diagonal jitter is added for finite-precision stability.

Relation to the empirical hierarchy. The hyperbolicity analysis in Fig. 1 motivates using Lorentz geometry to model the hierarchical organization of video tokens. Following standard practice in hyperbolic representation learning, HypToC maps vision tokens from the tangent space at the canonical Lorentz origin to the hyperboloid through the exponential map, and treats the origin as the root when computing continuous LCA depth. Rather than explicitly reconstructing a discrete empirical tree, HypToC uses the induced Lorentz geometry as a fixed, training-free geometric parameterization. Its practical contribution is isolated through the scale-matched Euclidean controls in Table 3 and the Euclidean segment-depth comparison in Table 15.

A.3 LCA Computation

In a rooted tree, the lowest common ancestor (LCA) of two nodes is the root of the smallest subtree containing both nodes. Its depth reflects their hierarchical proximity, with a deeper LCA indicating a smaller cluster and a more specific relationship. Chami et al. (2020) extend this discrete notion to continuous hyperbolic embeddings by treating the origin as the root and defining the LCA of two points as the point on their geodesic that is closest to the origin. This gives a rooted geometric affinity: if the closest point remains far from the origin, the pair behaves as if it shared a deeper continuous ancestor in the induced geometry. We use the same curvature $\kappa < 0$ for token Lorentz embedding and LCA computation, and denote the Lorentz origin by $\mathbf{o}_\kappa = [1/\sqrt{-\kappa}, 0, \ldots, 0]^\top$.

For two hyperbolic embeddings $\mathbf{z}_p, \mathbf{z}_q \in \mathbb{H}^m_\kappa$, their continuous LCA is defined as the point on the geodesic between them that is closest to the origin. Let

$$\Delta_{pq} = d_{\mathbb{H}_\kappa}(\mathbf{z}_p, \mathbf{z}_q), \qquad \eta_{pq} = \sqrt{-\kappa}\Delta_{pq}. \tag{23}$$

The unique geodesic from $\mathbf{z}_p$ to $\mathbf{z}_q$ can be parameterized by hyperbolic linear interpolation:

$$\gamma_{pq}(t) = \frac{\sinh((1-t)\eta_{pq})}{\sinh(\eta_{pq})}\mathbf{z}_p + \frac{\sinh(t\eta_{pq})}{\sinh(\eta_{pq})}\mathbf{z}_q, \qquad t \in [0,1]. \tag{24}$$

When $\eta_{pq} = 0$, Eq. 24 is interpreted by its limit, i.e., $\gamma_{pq}(t) = \mathbf{z}_p = \mathbf{z}_q$. The hyperbolic LCA is then

$$\mathbf{z}_p \vee \mathbf{z}_q = \gamma_{pq}(t^\star_{pq}), \qquad t^\star_{pq} = \arg\min_{t\in[0,1]} d_{\mathbb{H}_\kappa}(\mathbf{o}_\kappa, \gamma_{pq}(t)). \tag{25}$$

Because $d_{\mathbb{H}_\kappa}(\mathbf{o}_\kappa, \mathbf{z}) = \frac{1}{\sqrt{-\kappa}}\operatorname{arcosh}(\sqrt{-\kappa}z_0)$ on $\mathbb{H}^m_\kappa$, solving Eq. 25 is equivalent to minimizing the time coordinate of $\gamma_{pq}(t)$. Let

$$s_{pq} = \frac{(\mathbf{z}_p)_0\cosh(\eta_{pq}) - (\mathbf{z}_q)_0}{(\mathbf{z}_p)_0\sinh(\eta_{pq})}. \tag{26}$$

This gives the following numerically stabilized closed form:

$$t^\star_{pq} = \left[\frac{1}{\eta_{pq}}\operatorname{artanh}\left([s_{pq}]^{1-\epsilon}_{-1+\epsilon}\right)\right]^1_0, \tag{27}$$

where $[x]^b_a = \min(\max(x,a),b)$. For numerical stability, when η_{pq} is close to zero, we set $t^\star_{pq} = 1/2$. The LCA depth is then

$$\ell_{pq} = d_{\mathbb{H}_\kappa}(\mathbf{o}_\kappa, \mathbf{z}_p \vee \mathbf{z}_q) = \frac{1}{\sqrt{-\kappa}}\operatorname{arcosh}\left(\sqrt{-\kappa}(\mathbf{z}_p \vee \mathbf{z}_q)_0\right). \tag{28}$$

Thus ℓ_{pq} is not an additional learned score, but a scalar affinity induced by the fixed Lorentz geometry: larger values indicate a stronger continuous-LCA relation. HypToC uses this value as the edge weight for clustering non-anchor tokens.

Cross-frame object consistency. To quantitatively validate Fig. 1(c), we annotate object trajectories on 50 randomly sampled VideoMME videos, yielding 218 object tracks that span at least three sampled frames. A vision token is associated with an object when the center of its spatial patch falls inside the corresponding object region. We construct same-object pairs from tokens belonging to the same tracked object across frames and sample different-object pairs from the same videos with matched frame gaps. We sample 6,000 pairs from each group and repeat the sampling three times. Same-object pairs achieve an average LCA depth of 0.74 ± 0.02, compared with 0.36 ± 0.02 for different-object pairs, with an ROC-AUC of 0.82 ± 0.03. These results quantitatively support the cross-frame semantic consistency captured by continuous LCA depth.

B More Experimental Results

B.1 Baseline Comparability

Our controlled comparisons prioritize training-free methods whose compression boundary and token-budget definition can be aligned with HypToC. Compared methods use matched frame sampling, vision tokenization, benchmark settings, and retained vision-token budgets. AOT satisfies these conditions at the post-encoder compression stage, and we therefore reimplement it on Qwen2.5-VL and Qwen3-VL under the same evaluation protocol.

Some related methods optimize different stages of the Video-LLM pipeline or assume additional input information. For example, EarlyTom compresses tokens within the vision encoder, whereas MeToM additionally exploits codec metadata and performs multi-stage compression. For methods whose published settings differ in compression boundary, backbone, frame sampling, benchmark composition, or token-budget definition, we discuss them in Sec. 2 rather than directly mixing their reported numbers with our controlled comparisons.

Table 7: Additional comparison of training-free token compression methods on Qwen2.5-VL across retention ratios. AOT results are obtained from reimplementation under the same protocol.

Method	Retention Ratio R	VideoMME				EgoSchema		LongVideo Bench	MVBench	Avg.	
		Short	Medium	Long	Overall	Subset	Total			Score	Rel. Acc (%)
Qwen2.5-VL											
Full Tokens	100%	72.6	61.4	49.9	61.3	60.2	58.3	58.9	68.0	61.6	100.0
FastV	25%	71.2	57.8	**51.6**	60.2	**60.2**	57.2	54.6	**67.4**	59.8	97.1
VisionZip		70.7	57.9	51.2	59.9	58.6	57.0	57.1	67.3	60.3	97.9
FastVID		71.2	57.8	49.9	59.6	58.2	56.7	58.0	65.5	60.0	97.4
AOT		69.9	55.3	49.1	58.1	59.2	57.8	56.8	66.8	59.9	97.2
FlashVID		71.1	58.7	49.1	59.6	**59.4**	57.2	58.1	67.1	60.5	98.2
HypToC		**71.7**	**59.2**	51.0	**60.6**	58.2	**57.6**	**58.3**	**67.5**	**61.0**	**99.0**
FastV	15%	67.4	55.2	**51.1**	57.9	**59.6**	56.5	52.2	65.9	58.1	94.3
VisionZip		68.8	56.7	49.1	58.2	56.2	55.7	56.3	66.0	59.0	95.8
FastVID		68.2	56.9	49.4	58.2	56.6	56.0	56.8	63.6	58.6	95.1
AOT		69.2	54.1	48.6	57.3	58.9	57.3	54.3	66.7	58.9	95.6
FlashVID		69.8	57.1	49.3	58.7	57.6	56.4	56.8	66.6	59.6	96.8
HypToC		**70.6**	**57.7**	**50.8**	**59.7**	59.4	**57.2**	**57.7**	**67.2**	**60.5**	**98.1**

B.2 Additional Results on Qwen Models

Sec. 5.2 and Tab. 1 report the main comparisons on Qwen2.5-VL and Qwen3-VL. In addition to the 20% and 10% settings in the main text, Tab. 7 extends the AOT comparison to $R \in \{25\%, 15\%\}$ on Qwen2.5-VL. We reimplement AOT from its released implementation while following the same frame sampling, vision tokenization, benchmark evaluation, and output-token budgets used for the other methods. HypToC consistently achieves the highest average score across all evaluated retention ratios.

B.3 Additional Results on LLaVA Models

Sec. 5.2 and Tab. 2 present the main comparisons on LLaVA-OneVision and LLaVA-Video at $R \in \{20\%, 10\%\}$. Tab. 8 extends the evaluation to $R \in \{25\%, 15\%\}$ for both models. On LLaVA-OneVision, HypToC achieves average scores of 59.1 and 58.8 at the two additional ratios, corresponding to 101.2% and 100.7% relative accuracy. On LLaVA-Video, it obtains the highest average scores of 59.7 and 59.8, while reaching 60.2 on LongVideoBench at $R = 15\%$. These results complement the main comparison and show consistent performance across a broader range of token budgets.

B.4 Significance Analysis

HypToC is fully deterministic and involves neither stochastic training nor sampling during token compression. Given the same input frames and hyperparameters, both Intra-Frame Lorentz Selection and Inter-Frame Lorentz Clustering produce identical compressed tokens. All evaluations further use deterministic decoding, and therefore no training- or decoding-seed variation is involved.

We assess statistical significance using a video-level paired permutation test between HypToC and FlashVID. Because multiple evaluation questions can originate from the same video, we treat each video as a permutation block to preserve correlations among questions sharing the same visual input. Under the null hypothesis that the two methods perform equivalently, all paired predictions associated with a video are jointly exchanged between HypToC and FlashVID with probability 0.5. For each permutation, we recompute the accuracy on each benchmark and use the difference in the macro-average score over VideoMME, EgoSchema, LongVideoBench, and MVBench as the test statistic. We perform $10{,}000$ permutations for each model and retention ratio and compute a two-sided p-value as the proportion of permutations whose absolute score difference is at least as large as the observed difference.

Table 8: Additional comparison of training-free token compression methods on LLaVA-OneVision and LLaVA-Video across retention ratios.

Method	Retention Ratio R	VideoMME				EgoSchema		LongVideo Bench	MVBench	Avg.	
		Short	Medium	Long	Overall	Subset	Total			Score	Rel. Acc (%)
LLaVA-OneVision											
Full Tokens	100%	69.9	56.7	48.9	58.5	62.2	60.3	56.6	58.3	58.4	100.0
FastV	25%	68.1	54.7	46.8	56.5	60.4	57.8	55.4	56.4	56.5	96.7
VisionZip		68.8	57.3	48.2	58.1	63.0	60.4	56.4	57.8	58.2	99.7
PruneVID		67.3	54.8	47.2	56.4	61.0	58.1	55.4	56.8	56.7	97.1
FastVID		69.9	56.3	47.4	57.9	61.2	59.5	55.9	58.1	57.8	99.0
FlashVID		71.2	57.0	**49.3**	59.2	63.4	60.4	56.8	58.0	58.6	100.3
HypToC		**72.7**	**57.9**	48.9	**59.5**	**63.8**	**60.6**	**58.0**	**58.4**	**59.1**	**101.2**
FastV	15%	64.6	54.0	45.3	54.6	59.8	56.6	54.8	55.0	55.2	94.5
VisionZip		63.8	54.6	48.3	55.6	62.8	60.0	54.1	53.5	55.8	95.5
FastVID		69.7	55.8	47.7	57.7	58.8	58.9	56.7	**58.2**	57.9	99.1
PruneVID		67.2	52.8	46.7	56.1	61.6	57.7	54.5	55.1	55.7	95.4
FlashVID		69.6	56.0	**48.9**	58.2	62.8	60.4	57.5	57.9	58.5	100.2
HypToC		**70.0**	**58.0**	47.8	**58.6**	**63.2**	**60.9**	**57.8**	57.9	**58.8**	**100.7**
LLaVA-Video											
Full Tokens	100%	77.0	62.1	53.3	64.2	59.4	57.3	59.5	61.9	60.7	100.0
FastV	25%	71.7	59.2	50.9	60.6	56.0	54.8	56.4	59.1	57.7	95.1
VisionZip		74.0	60.3	52.9	62.4	59.0	**57.0**	58.3	60.0	59.4	97.9
FastVID		74.7	60.1	**53.6**	**62.8**	57.4	55.4	58.2	**60.5**	59.2	97.5
FlashVID		74.2	61.4	51.6	62.4	**59.2**	56.6	59.1	60.2	59.6	98.2
HypToC		**75.3**	**62.3**	50.2	62.6	58.2	56.3	**59.7**	60.3	**59.7**	**98.4**
FastV	15%	67.9	56.9	50.8	58.5	52.8	53.1	54.5	57.5	55.9	92.1
VisionZip		72.9	58.1	51.9	61.0	58.6	55.7	57.2	59.6	58.4	96.2
FastVID		73.4	58.1	51.8	61.1	56.8	54.1	57.7	60.3	58.3	96.0
FlashVID		73.8	59.6	52.1	61.8	57.8	55.8	58.3	**60.4**	59.1	97.4
HypToC		**74.0**	**60.6**	**52.3**	**62.3**	**58.8**	56.7	**60.2**	60.1	**59.8**	**98.5**

Since the analysis contains 16 model-budget comparisons, we additionally apply the Benjamini–Hochberg procedure to control the false discovery rate at 0.05. Table 9 reports both the raw and BH-adjusted p-values, and statistical significance is determined using the adjusted p-values.

As shown in Table 9, the improvements remain statistically significant after video-level permutation and multiple-comparison correction in 14 of the 16 evaluated model-budget settings. All results on Qwen3-VL and Qwen2.5-VL remain significant. On the LLaVA models, only the two smallest improvements of $+0.1$ average points, namely LLaVA-OneVision at $R = 20\%$ and LLaVA-Video at $R = 25\%$, do not reach the corrected significance threshold. Overall, the analysis confirms that the performance gains are statistically reliable across the vast majority of evaluated settings, while avoiding overinterpretation of negligible score differences.

B.5 Computation Efficiency

Implementation Details. *(i) Efficient implementation.* HypToC is implemented with efficient frame-wise and blockwise operations. For Intra-Frame Lorentz Selection, pairwise Lorentz distances and the frame-wise distance are computed independently for each frame. The upper triangular distance matrix is mirrored to construct a symmetric similarity matrix, followed by a small diagonal jitter. The greedy k-DPP solver maintains incremental Cholesky factors, avoiding repeated determinant computation during selection. For Inter-Frame Lorentz Clustering, LCA depths are evaluated in 512×512 token blocks using the closed-form computation in Appendix A.3 and vectorized GPU operations. A streaming buffer initially retains the highest-scoring $8|\mathcal{U}|$ token pairs, avoiding materialization and global sorting of the complete residual-token pair matrix. If the buffered pairs are insufficient to reduce the number of connected components to K_m, we double the buffer capacity

Table 9: Video-level paired significance analysis between HypToC and FlashVID. Δ denotes the difference in average score. Raw p-values are obtained from 10,000 video-level paired permutations and adjusted over the 16 comparisons using the Benjamini–Hochberg procedure. Bold adjusted p-values indicate statistical significance at $p < 0.05$.

Model	R	**FlashVID**	**HypToC**	Δ	**Raw** p	**BH-adj.** p
Qwen3-VL	25%	63.0	64.2	+1.2	< 0.001	$< \mathbf{0.001}$
	20%	62.2	63.6	+1.4	< 0.001	$< \mathbf{0.001}$
	15%	61.2	63.0	+1.8	< 0.001	$< \mathbf{0.001}$
	10%	60.3	61.9	+1.6	< 0.001	$< \mathbf{0.001}$
Qwen2.5-VL	25%	60.5	61.0	+0.5	0.012	**0.017**
	20%	60.2	60.9	+0.7	0.006	**0.014**
	15%	59.6	60.5	+0.9	0.002	**0.006**
	10%	58.9	59.7	+0.8	0.004	**0.011**
LLaVA-OneVision	25%	58.6	59.1	+0.5	0.010	**0.016**
	20%	58.7	58.8	+0.1	0.091	0.097
	15%	58.5	58.8	+0.3	0.016	**0.021**
	10%	57.9	58.1	+0.2	0.021	**0.024**
LLaVA-Video	25%	59.6	59.7	+0.1	0.108	0.108
	20%	59.3	59.8	+0.5	0.018	**0.022**
	15%	59.1	59.8	+0.7	0.008	**0.014**
	10%	58.2	58.9	+0.7	0.007	**0.014**

and continue the blockwise top-pair extraction until exactly K_m clusters are obtained. This adaptive expansion preserves the exact output-token budget while reducing working memory.

(ii) Exact budget allocation. The total output budget is fixed as $K = \lfloor RM \rfloor$. The anchor budget is distributed uniformly across frames using $k_s = \lfloor \alpha K/F \rfloor$, yielding $K_s = Fk_s$ anchor tokens, while the remaining $K_m = K - K_s$ slots are assigned to merged tokens. Therefore, the final sequence always contains exactly K vision tokens. At $\alpha = 1$, integer allocation may leave a small remainder $K_m = K - F\lfloor K/F \rfloor < F$ for clustering. For the default Qwen3-VL setting with $F = 32$ and $K = 627$, this corresponds to $K_s = 608$ anchor tokens and $K_m = 19$ merged tokens.

(iii) Compression interface. For all evaluated models, HypToC operates on the model-native projected vision tokens after the vision projection stage, including the model-native token merger when present, and before the resulting vision-token sequence is passed to the language model. The original vision encoder, projector, and language model remain unchanged.

(iv) Position handling. For Qwen2.5-VL and Qwen3-VL, each retained anchor preserves its original 3D MRoPE position identifiers, while each merged token inherits the complete temporal and spatial position identifiers of its cluster center. For LLaVA-OneVision and LLaVA-Video, compressed tokens are first ordered according to their inherited spatiotemporal positions, after which the model's standard sequential position identifiers are assigned.

(v) Attention extraction. Token saliency is obtained from the final vision self-attention layer and averaged across attention heads. When no dedicated `[CLS]` token is available, the attention received by each token is averaged over all query tokens within the frame. Only the required vision-layer attention is materialized; HypToC does not access language-model attention matrices.

Runtime Breakdown. We profile the default Qwen3-VL setting with $F = 32$ frames and an average of $N = 196$ vision tokens per frame, corresponding to $M = 6{,}272$ input vision tokens. At $R = 10\%$ and $\alpha = 0.8$, $K_s = 480$ anchor tokens are retained and $|\mathcal{U}| = 5{,}792$ residual tokens are processed into $K_m = 147$ merged tokens. The clustering stage evaluates approximately 16.8 million residual-token pairs through blockwise GPU computation.

Table 10 reports the runtime of individual components. All latency measurements are averaged over 100 measured runs after 20 warm-up runs. The total compression latency is 44.8 ± 1.4 ms. LCA

clustering constitutes the largest fraction of the compression stage, while Lorentz projection and scale-calibrated kernel construction remain lightweight.

Table 10: Runtime breakdown of HypToC on Qwen3-VL at $R = 10\%$. Latency is averaged over 100 measured runs after 20 warm-up runs.

Component	Latency (ms)	Fraction
Lorentz projection and scale-calibrated kernel	4.5	10.0%
Greedy k-DPP selection	9.4	21.0%
LCA clustering	24.0	53.6%
Center-aware aggregation	6.9	15.4%
Total	**44.8**	**100.0%**

End-to-End Efficiency. We compare HypToC with FlashVID on Qwen3-VL under exactly matched vision-token budgets and additionally report the full-token model as the uncompressed reference. At each retention ratio, FlashVID and HypToC pass the same number K of vision tokens to the frozen LLM and therefore incur identical LLM computation. Vision encoding is shared by all settings and takes 512.6 ms on average. It is included in TTFT but omitted from the reported prefilling time. Latency is averaged over 100 measured runs after 20 warm-up runs. The standard deviation of compression latency ranges from 1.3 to 1.9 ms across the reported settings.

Table 11: Efficiency comparison on Qwen3-VL under matched vision-token budgets. Compression latency, LLM forward time, prefilling time, and TTFT are reported in milliseconds. LLM TFLOPs denote the computation of the frozen language model, while Peak Mem. denotes additional GPU memory during token compression.

Method	R	LLM TFLOPs	Compression	LLM Forward	Prefill	TTFT	Peak Mem.	Avg. Score
Full Tokens	100%	108.4	–	1018.6	1018.6	1531.2	–	63.9
FlashVID	25%	27.4	58.7	267.4	326.1	838.7	1.14 GB	63.0
HypToC	25%	27.4	53.6	267.4	321.2	833.8	1.08 GB	**64.2**
FlashVID	20%	22.1	54.8	228.6	283.4	796.0	1.12 GB	62.2
HypToC	20%	22.1	50.2	228.6	278.8	791.4	1.08 GB	**63.6**
FlashVID	15%	16.8	51.6	190.2	241.8	754.4	1.09 GB	61.2
HypToC	15%	16.8	47.3	190.2	237.5	750.1	1.09 GB	**63.0**
FlashVID	10%	11.5	48.9	153.6	202.5	715.1	1.07 GB	**60.3**
HypToC	10%	11.5	44.8	153.6	198.4	711.0	1.09 GB	**61.9**

The compression-stage computation is small relative to the subsequent language-model forward pass. Across $R = 25\%$–10%, FlashVID requires approximately 0.62–0.74 TFLOPs for token compression, while HypToC requires 0.70–0.81 TFLOPs. These costs are reported separately from the LLM TFLOPs in Table 11. Under matched output-token budgets, HypToC maintains compression latency and peak memory comparable to FlashVID while consistently achieving higher average accuracy. Relative to the full-token model, at $R = 10\%$ HypToC reduces prefilling time from 1018.6 ms to 198.4 ms and TTFT from 1531.2 ms to 711.0 ms, corresponding to approximately $5.1\times$ and $2.2\times$ speedups, respectively.

Scaling with Video Length. We further evaluate computational scaling by varying the number of sampled frames while keeping the per-frame token number $N = 196$, $R = 10\%$, and $\alpha = 0.8$ fixed. Table 12 reports the resulting input and residual token scales, pair counts, compression latency, and peak GPU memory.

The blockwise implementation maintains moderate memory growth as the number of frames increases. At 64 frames, corresponding to more than 12K original vision tokens, HypToC processes approximately 67.1M residual-token pairs with 127.7 ms compression latency and 1.19 GB peak compression memory. These results show that the proposed implementation scales to the tested setting of 64 sampled frames without materializing the complete pairwise relation matrix.

Table 12: Scaling analysis on Qwen3-VL at $R = 10\%$. The per-frame token number is fixed at $N = 196$.

Frames	Input Tokens	Residual Tokens	Token Pairs	Compression Latency (ms)		Peak Memory (GB)	
F	M (K)	$\lvert\mathcal{U}\rvert$ (K)	(M)	FlashVID	HypToC	FlashVID	HypToC
8	1.57	1.45	1.0	12.5	11.0	0.95	0.91
16	3.14	2.90	4.2	21.7	20.4	1.00	0.95
32	6.27	5.79	16.8	48.9	44.8	1.12	1.06
64	12.54	11.58	67.1	136.2	127.7	1.28	1.19

B.6 Lorentz Geometry Sensitivity

Unless otherwise specified, we use $\alpha = 0.8$, $\kappa = -1$, $\rho = 0.02$, $\tau_s = 1$, $\tau_m = 1$, and $\epsilon = 10^{-6}$ for all models, benchmarks, and retention ratios. Both τ_s and τ_m are dimensionless because the corresponding Lorentz distances are normalized by their respective distance scales.

We analyze sensitivity to the sectional curvature κ and token scaling factor ρ on Qwen3-VL at $R = 10\%$. We evaluate $\kappa \in \{-2, -1, -0.5\}$ and $\rho \in \{0.01, 0.02, 0.05\}$. During each sweep, the other parameter is fixed at its default value.

Table 13: Sensitivity to Lorentz curvature κ and radial scale ρ on Qwen3-VL at $R = 10\%$. The other parameter is fixed at its default value during each sweep.

Parameter	Value	VideoMME	EgoSchema	LongVideo Bench	MVBench	Rel. Acc.
κ	-2.0	60.4	65.7	56.9	62.3	96.2
	$\mathbf{-1.0}$	**61.0**	**66.0**	**57.5**	**62.9**	**96.9**
	-0.5	60.8	65.4	57.3	62.7	96.4
ρ	0.01	60.8	65.9	57.4	62.7	96.6
	0.02	**61.0**	**66.0**	**57.5**	**62.9**	**96.9**
	0.05	60.7	65.7	57.2	62.5	96.3

Table 14: Ablation of token magnitude and direction in Lorentz mapping on Qwen3-VL at $R = 10\%$.

Variant	VideoMME	EgoSchema	LongVideoBench	MVBench	Rel. Acc.
Constant norm	60.4	65.2	56.9	62.2	95.8
Norm-shuffled	60.1	65.0	56.8	62.0	95.4
Direction-shuffled	58.1	63.3	55.3	60.4	92.8
HypToC	**61.0**	**66.0**	**57.5**	**62.9**	**96.9**

Table 14 separates the contributions of token magnitude and direction. For the constant norm control, each token direction is rescaled to the median norm of the original token set. Both magnitude and direction contribute to Lorentz mapping. Removing or shuffling token magnitudes degrades performance, while shuffling token directions causes a larger drop. The full mapping performs best, indicating that its gains cannot be explained by token norm or directional information alone.

For the E-DPP and L-DPP comparison in Table 3, Euclidean and Lorentz pairwise distances are normalized by their respective frame-wise distances before kernel construction. Specifically, E-DPP uses $\sigma_f^{\mathbb{E}} = \mathrm{median}_{i<j} \|\mathbf{x}_{f,i} - \mathbf{x}_{f,j}\|_2$, while L-DPP uses the Lorentz counterpart $\sigma_f^{\mathbb{H}}$. Both variants use the same dimensionless temperature $\tau_s = 1$, as well as identical token quality, selection budget, and MAP solver. Therefore, their comparison isolates the choice of geometry rather than the absolute distance scale.

B.7 Ablation Analysis of Inter-Frame Lorentz Clustering

Cross-Frame Clustering Behavior. We first examine whether the resulting clusters indeed aggregate information across frames. At $R = 10\%$, 78.4% of non-singleton clusters contain tokens from multiple frames, covering 3.6 distinct frames on average. Their average normalized temporal

span, $(\max_{p\in C_g} f_p - \min_{p\in C_g} f_p)/(F-1)$, is 0.41. Restricting clustering to same-frame pairs reduces relative accuracy to 95.3%, while using only cross-frame pairs achieves 96.2%, compared with 96.9% for unrestricted clustering. This shows that cross-frame relations provide the major contribution while retaining complementary within-frame evidence. On four chronology-sensitive MVBench categories, HypToC further achieves an average score of 64.1, compared with 62.7 for FlashVID, indicating that cross-frame aggregation remains compatible with order-sensitive understanding.

On Clustering and Aggregation. We further ablate the two components of Inter-Frame Lorentz Clustering: how residual tokens are clustered and how tokens within each cluster are aggregated. For the clustering ablation, we replace the proposed LCA-depth clustering with several alternatives while keeping Intra-Frame Lorentz Selection, the token budget, and subsequent aggregation unchanged. *Euclidean distance*, *Cosine similarity*, and *Lorentz distance* use the corresponding pairwise measure to progressively merge the most related residual tokens. *Euclidean segment depth* is the Euclidean analogue of LCA depth, defined as the minimum distance from the origin to the line segment connecting a token pair. We also compare with *Lorentz k-medoids*, which clusters the mapped tokens directly using Lorentz distance. For the aggregation ablation, we use the proposed LCA clustering and replace the center-aware aggregation with different alternatives.

Table 15: Ablation of Inter-Frame Lorentz Clustering on Qwen3-VL at $R = 10\%$.

Variant	VideoMME	EgoSchema	LongVideoBench	MVBench	Rel. Acc.
Clustering method					
Euclidean distance	59.9	64.7	56.7	61.9	95.2
Cosine similarity	60.2	65.0	56.9	62.1	95.6
Euclidean segment depth	60.5	65.4	57.1	62.4	96.0
Lorentz distance	60.5	65.4	57.2	62.5	96.2
Lorentz k-medoids	60.6	65.1	56.9	62.2	95.8
LCA depth (Ours)	**61.0**	**66.0**	**57.5**	**62.9**	**96.9**
Token aggregation					
Mean pooling	60.1	64.8	56.8	62.0	95.4
Center token	60.4	65.1	57.0	62.2	95.8
Quality only	60.6	65.4	57.1	62.3	96.1
Center distance only	60.5	65.3	57.2	62.5	96.1
Quality + center distance (Ours)	**61.0**	**66.0**	**57.5**	**62.9**	**96.9**

As shown in Table 15, the proposed LCA-depth clustering consistently outperforms Euclidean, cosine, Lorentz-distance, and Lorentz k-medoids alternatives. The improvement over pairwise Lorentz distance shows that the rooted LCA relation captures information beyond pairwise proximity. Its advantage over Euclidean segment depth further reflects the use of Lorentz geodesics rather than Euclidean line segments. For token aggregation, the complete quality- and center-aware weighting also performs better than mean pooling, direct center-token retention, or either weighting component alone.

Transferability of the Anchor Ratio. Table 16 evaluates α across representative model families and token budgets. Entries report the average score over the four benchmarks.

Table 16: Transferability of α across models and token budgets.

Model	R	$\alpha = 0.6$	$\alpha = 0.8$	$\alpha = 1.0$
Qwen3-VL	20%	63.2	**63.6**	62.9
Qwen3-VL	10%	61.7	**61.9**	61.3
LLaVA-Video	20%	59.5	**59.8**	59.2
LLaVA-Video	10%	58.6	**58.9**	58.4

The same $\alpha = 0.8$ setting remains effective across the tested model families and retention ratios, supporting the transferability of the selection-clustering balance without model-specific tuning.

C Additional Qualitative Analysis

C.1 Additional Intra-Frame Token Selection Results

Sec. 5.3 and Fig. 3 visualize how Intra-Frame Lorentz Selection differs from the selection strategy in FlashVID (Fan et al., 2026). Fig. 5 provides additional examples on Qwen3-VL. Although both methods retain many common tokens, those selected by HypToC more frequently correspond to people, facial details, textual content, foreground objects, and action-related regions. These examples further illustrate that HypToC preserves fine-grained semantic evidence while maintaining broader coverage of the frame content.

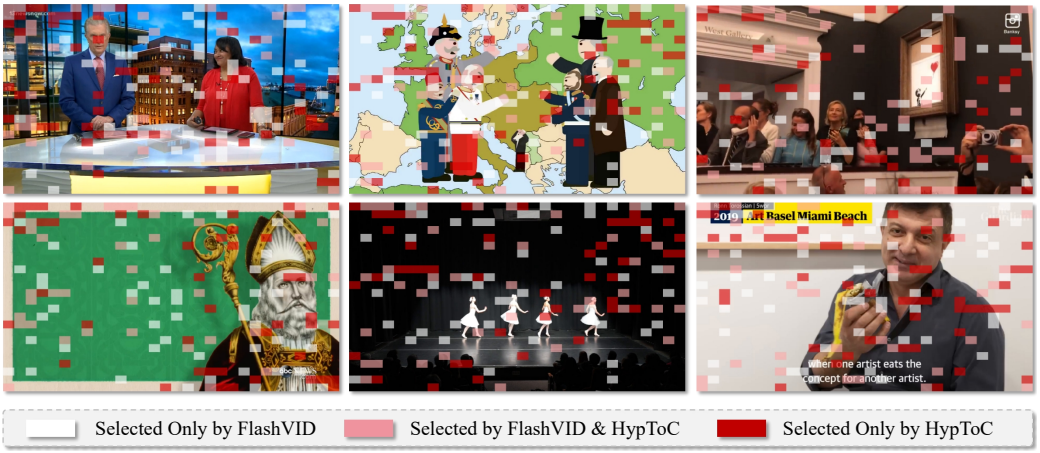


Figure 5: **Additional visualization of intra-frame token selection on Qwen3-VL.** White regions denote tokens selected only by FlashVID, pink regions denote tokens selected by both methods, and red regions denote tokens selected only by HypToC. HypToC additionally retains tokens associated with subjects, facial details, text, foreground objects, and action-related content.

C.2 Additional Vision Token Norm Distribution Results

The main text reports the token-norm distribution on Qwen3-VL in Fig. 4. Fig. 6 extends this analysis to VideoMME, EgoSchema, LongVideoBench, and MVBench, respectively. Across all four benchmarks, HypToC-selected tokens exhibit distributions closer to those of the full token sets than FlashVID-selected tokens. This consistent pattern shows that Intra-Frame Lorentz Selection preserves the original vision token norm statistics across diverse video domains and reasoning tasks.

D Case Studies

We present case studies on Qwen3-VL and LLaVA-Video to complement the quantitative results. The examples cover VideoMME, EgoSchema, LongVideoBench, and MVBench, including temporal ordering, multi-step action understanding, scene-transition tracking, and long-range semantic reasoning. In each figure, red and green text denote incorrect predictions from the original model and correct predictions obtained with HypToC, respectively.

D.1 Case Studies on Qwen3-VL

Fig. 7 presents examples from VideoMME and EgoSchema. HypToC correctly recovers the chronological order of activities, identifies the group of a fallen athlete, and selects the statement supported by the historical video. For procedural questions, it also captures more complete action sequences, such as removing plants, creating artwork, and preparing paste in a blender. These cases require evidence distributed across multiple frames rather than recognition from an isolated image.

Fig. 8 shows examples from LongVideoBench and MVBench. HypToC more accurately identifies scene transitions and events occurring before or after a referenced moment. For example, it captures

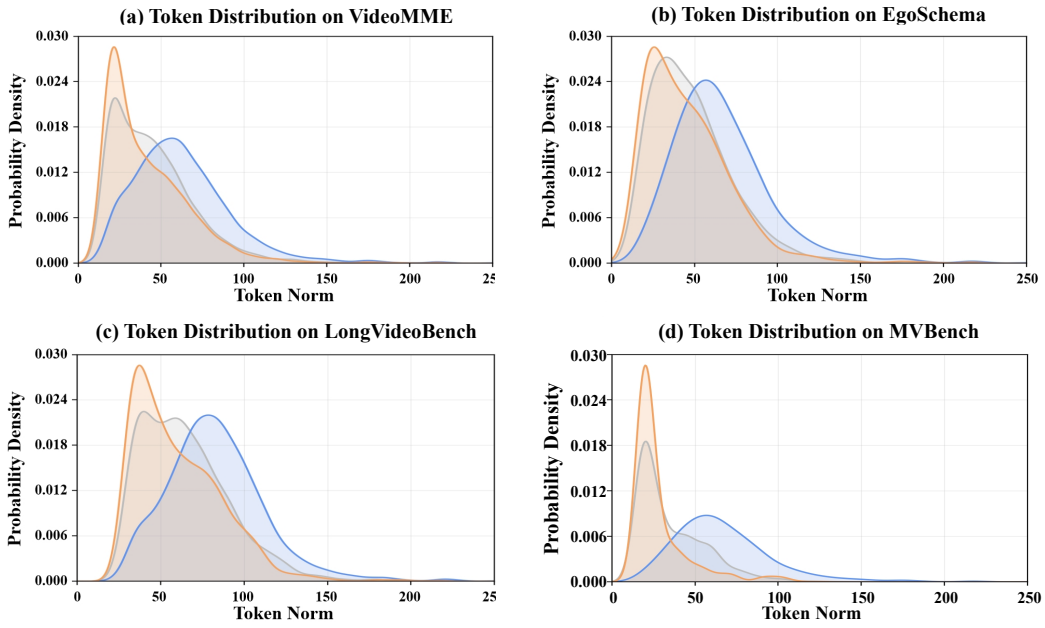


Figure 6: **Vision token norm distributions on four video benchmarks using Qwen3-VL.** The gray, blue, and orange curves represent all tokens, FlashVID-selected tokens, and HypToC-selected tokens, respectively. HypToC more closely follows the full-token norm distribution on VideoMME, EgoSchema, LongVideoBench, and MVBench.

the car skidding after the driver turns the steering wheel, recognizes the season following the red-leaf forest scene, and distinguishes transitions between indoor and outdoor locations. The results illustrate improved retention of temporally separated evidence in long and dynamically changing videos.

D.2 Case Studies on LLaVA-Video

Fig. 9 presents VideoMME and EgoSchema examples on LLaVA-Video. HypToC correctly infers the relation between the Triangulum and Andromeda galaxies, recovers the country order in a relay race, and distinguishes contrasting views on Bitcoin. It also retains fine-grained procedural information when describing knife sharpening, pottery making, and painting-tool maintenance. These examples show that HypToC preserves both semantic concepts and detailed action sequences across diverse domains.

Fig. 10 further reports LongVideoBench and MVBench examples. HypToC accurately follows transitions between locations and scenes, including changes from a security room to a corridor, from the ground to a rooftop, and from a road to a helicopter view. It also resolves a temporally grounded question concerning an object absent from a referenced scene. These cases indicate that the compressed tokens preserve scene-level context and temporal dependencies across distant frames.

D.3 Failure Case Analysis

Fig. 11 presents failure cases from VideoMME, EgoSchema, and MVBench. The first case requires identifying the primary distinction among several valid details presented in the video. This depends on semantic emphasis rather than the spatial or temporal hierarchy. In the second case, motion blur obscures the actions performed on the sacks. In the third case, defocus blur weakens the scene cues required to recognize the transition. In the fourth case, low illumination suppresses important visual details across frames. Semantic prioritization and video restoration remain potential directions for future research.

E USAGE OF LARGE LANGUAGE MODELS

In this work, Large Language Models (LLMs) are only used for polishing the paper writing. They are not involved in research ideation, experimental design, data analysis, or the formulation of conclusions. All substantive intellectual contributions are made by the authors. The authors have checked the content throughout the paper after polishing, and take full responsibility for the current form.



Question: *Which of the following is the correct chronological order of what Bart does on his first day in the video?*

Qwen3-VL: *Breakfast , gym workout, bison watching.* ✗ **HypToC:** *Gym workout, breakfast, bison watching.* ✓



Question: *To which group does the athlete who fell in the video belong?*

Qwen3-VL: *The man struggled in the trunk.* ✗ **HypToC:** *Second group.* ✓



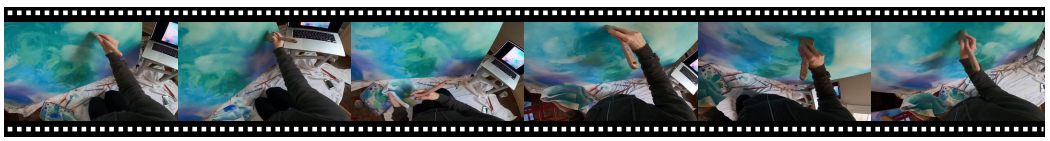
Question: *In line with the video evidence, which of the following statements about Napoleon is correct?*

Qwen3-VL: *His Jacobin dispute stalled his career.* ✗ **HypToC:** *His code denied wives any choice of residence.* ✓



Question: *Analyze and outline the person's approach to the primary activity shown in the video.*

Qwen3-VL: *Check the plants, then point at a plant.* ✗ **HypToC:** *Glove up, identify the plants, pull into a bucket.* ✓



Question: *Describe the key sequence of events taking place in this video that shows c's process of creating art.*

Qwen3-VL: *Examine the board, then wipes the brush.* ✗ **HypToC:** *Check the laptop, then paints with the brush.* ✓



Question: *What were the primary steps to prepare the paste in the blender before and after adding a new ingredient?*

Qwen3-VL: *Add powder, blend it, taste, add more.* ✗ **HypToC:** *Add powder, blend it, taste, turn off blender.* ✓

Figure 7: **Case studies on Qwen3-VL from VideoMME and EgoSchema.** HypToC improves temporal ordering, fine-grained event recognition, and multi-step action understanding. Red and green denote incorrect original predictions and correct HypToC predictions, respectively.



Question: *A window appears on the screen that reads 'Amanda Blanc'. What change did this woman experience?*

Qwen3-VL: *The woman took a sip of water.* ✗ **HypToC:** *The woman raised both hands.* ✓



Question: *What happened after the driver looked towards the rear window and turned the steering wheel?*

Qwen3-VL: *It cannot be inferred from this video.* ✗ **HypToC:** *A white car skidded on the dusty ground.* ✓



Question: *After a video scene of a woman walking through a forest filled with red leaves, which season appears first?*

Qwen3-VL: *Autumn.* ✗ **HypToC:** *Winter.* ✓



Question: *Based on the video, which choice shows the scene changes accurately?*

Qwen3-VL: *From the office to the parking lot.* ✗ **HypToC:** *From the elevator to the corner of the room.* ✓



Question: *What's the right option for how the scenes in the video change?*

Qwen3-VL: *From the city to the countryside.* ✗ **HypToC:** *From the land to the sea.* ✓



Question: *Which choice tells us the correct order of scenes in the video?*

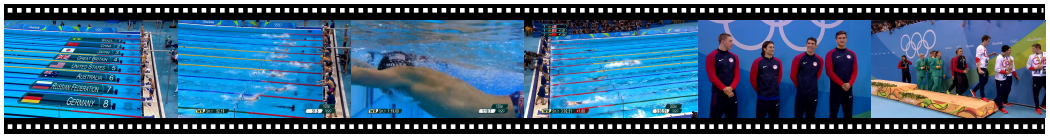
Qwen3-VL: *From the subway to the beach.2.* ✗ **HypToC:** *From the elevator to the front of the house.1.* ✓

Figure 8: **Case studies on Qwen3-VL from LongVideoBench and MVBench.** HypToC better captures event consequences, temporal references, and scene transitions over extended video contexts.



Question: *What can be inferred about the Triangulum Galaxy (M33) from the other two galaxies??*

LLaVA-Video: *It gravitationally interacts with M31.* ✗ **HypToC:** *Andromeda satellite galaxy.* ✓



Question: *What was the order of countries that finished first for each leg of the relay race in the video?*

LLaVA-Video: *USA, UK, UK, USA.* ✗ **HypToC:** *USA, UK, USA, USA.* ✓



Question: *Summarize the core difference between the contrasting viewpoints on the future of Bitcoin in the video.*

LLaVA-Video: *Bitcoin stability or returns.* ✗ **HypToC:** *Bitcoin integration or independence.* ✓



Question: *Describe the main steps that character c took to accomplish their objective.*

LLaVA-Video: *Pick up a knife and cut the wood.* ✗ **HypToC:** *Sharpen a knife on wood using a gadget.* ✓



Question: *Summarize how the character shapes and finishes the pottery.*

LLaVA-Video: *Centers, smooths, and cleans the pot.* ✗ **HypToC:** *Shapes, smooths, removes, and cleans the bowl.* ✓



Question: *How do her painting and tool care differ between the two halves?*

LLaVA-Video: *Paints the wall, and stores the brush.* ✗ **HypToC:** *Cleans, paints the desk, and stores the brush.* ✓

Figure 9: **Case studies on LLaVA-Video from VideoMME and EgoSchema.** HypToC improves semantic inference, chronological reasoning, and detailed procedural understanding. Red and green denote incorrect original predictions and correct HypToC predictions, respectively.



Question: *When 'Not the least of which, is' is mentioned, what object is not present in the scene?*

LLaVA-Video: *Flowers of yellow flower clusters.* ✗ **HypToC:** *Brown long-sleeved outfit.* ✓



Question: *Which choice tells us the correct order of scenes in the video?*

LLaVA-Video: *From the living room to the stairs.* ✗ **HypToC:** *From the security room to the corridor.* ✓



Question: *Based on the video, which choice shows the scene changes accurately?*

LLaVA-Video: *From the school to the government.* ✗ **HypToC:** *From the home to the Department of Education.* ✓



Question: *Which choice tells us the correct order of scenes in the video?*

LLaVA-Video: *From memories to the beaches.* ✗ **HypToC:** *From thoughts to the scene outside the prison.* ✓



Question: *Which choice matches the scene changes in the video?*

LLaVA-Video: *From the bedroom to the living room.* ✗ **HypToC:** *From the ground to the rooftop.* ✓



Question: *Which choice matches the scene changes in the video?*

LLaVA-Video: *From the city street to the airplane.* ✗ **HypToC:** *From the road to the helicopter above.* ✓

Figure 10: **Case studies on LLaVA-Video from LongVideoBench and MVBench.** HypToC improves scene-transition tracking and temporally grounded recognition across long video sequences.



Question: *What is the core distinction between American Express and companies like Visa in the video?*

Qwen3-VL: *AmEx lends, Visa only network.* ✓ **HypToC:** *AmEx rewards attract higher spenders than Visa.* ✗



Question: *Summarize how the actions with the first sack differ from those with the later sacks.*

Qwen3-VL: *He pours the first but opens the rest.* ✓ **HypToC:** *He places the first sack but folds the rest.* ✗



Question: *Based on the video, which choice shows the scene changes accurately?*

Qwen3-VL: *From Taylor's room to the parking lot.* ✓ **HypToC:** *From John's house to the city park.* ✗



Question: *Which choice matches the scene changes in the video?*

Qwen3-VL: *From the alley to the room.* ✓ **HypToC:** *From the street to the office.* ✗

Figure 11: **Failure cases on VideoMME, EgoSchema, and MVBench.** Green and red denote correct predictions from Qwen3-VL and incorrect predictions with HypToC, respectively.