STRUCTURE-PRESERVING VISUAL TOKEN COMPRESSION FOR STREAMING VIDEO UNDERSTANDING VIA BILATERAL GRID SPLATTING

Anonymous authors
Paper under double-blind review

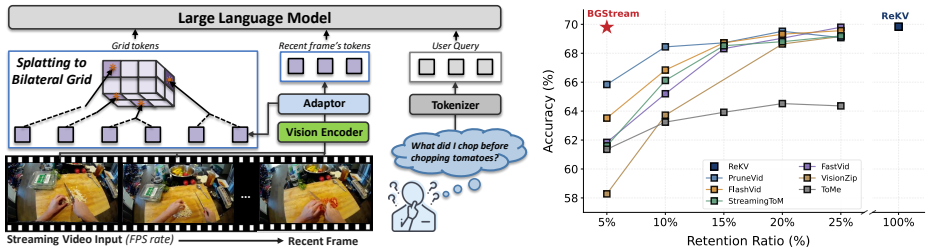


Figure 1: **Left**: BGStream is a training-free visual token compression method for streaming video understanding that splats tokens into the bilateral grid, preserving the semantic structure. **Right**: Performance on StreamingBench (Lin et al., 2024) across various retention ratios. ReKV (Di et al., 2025) is shown for reference; it optimizes the KV cache rather than compressing tokens.

ABSTRACT

Streaming video understanding requires Video MLLMs to process incoming frames in real time, but rapidly accumulating visual tokens incur a severe computational bottleneck. Visual token compression offers a remedy, yet existing methods fall short in the streaming setting on two fronts: the unbounded frame stream forces aggressive compression under which their accuracy collapses, and they require a temporal boundary prior unavailable in the streaming setting. To bridge this gap, we propose BGStream, a training-free compression method based on the *bilateral grid* that (i) preserves the semantic structure of visual features even under aggressive retention ratios, and (ii) splats each incoming frame's tokens into the grid on the fly, without accessing the full video. However, generalizing the bilateral grid to visual tokens is non-trivial; hence, we ground this design in a theoretical analysis based on the Johnson–Lindenstrauss lemma. On StreamingBench and OVO-Bench, BGStream outperforms prior compression methods while remaining compatible with KV cache optimization.

1 INTRODUCTION

Video Multimodal Large Language Models (Video MLLMs) (Zhang et al., 2024c; Shen et al., 2024) are typically designed for the *offline* setting, where the entire video and user query are jointly provided to the model. However, real-world applications such as augmented reality glasses (Wen et al., 2025) and live sports broadcasting (Rao et al., 2025) increasingly demand *streaming* video understanding. In this setting, the model must process incoming frames in real time at the FPS level, rather than operating on a fixed number of sampled frames, fundamentally departing from the offline paradigm.

Despite its promising applications, streaming video understanding with Video MLLMs remains challenging due to the heavy computational cost of processing a large number of visual tokens (Chen et al., 2025). As frames continuously arrive and visual tokens rapidly accumulate, the LLM is burdened with performing attention computation over an ever-growing visual token sequence. Prior streaming approaches attempt to mitigate this cost by offloading or compressing the KV cache within the LLM (e.g., ReKV (Di et al., 2025)). However, these methods fail to address the root cause of the bottleneck, as visual tokens still pass through the LLM layers and incur the full attention cost.

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.

To tackle this, visual token compression has emerged as a promising approach that reduces visual tokens before they enter the LLM (Fan et al., 2026). However, existing methods fall short when adapted to streaming understanding for two reasons. (1) They struggle under the aggressive retention ratios[1] (see Figure 1(Right)); yet such ratios are unavoidable in streaming, where unbounded frame arrival makes visual tokens accumulate beyond the LLM's context budget. (2) They presuppose a temporal boundary prior over the entire video; yet such a prior is unavailable in streaming, where future frames have not yet arrived.

To bridge this gap, we propose **B**ilateral **G**rid-based **Stream**ing compression (BGStream), a training-free visual token compression method based on the *bilateral grid*, a structured representation originally developed for edge-preserving image filtering (Chen et al., 2007). The bilateral grid offers two properties that address the two limitations above. First, it preserves the semantic structure of visual tokens even at coarse resolutions—e.g., an 8-megapixel image can be faithfully represented by a grid of only 2K cells (Chen et al., 2016)—enabling robust compression under aggressive retention ratios that streaming forces. Second, its *splatting* operation additively accumulates each visual tokens into a grid cell, allowing incoming frames to be incorporated on the fly without access to the full video. Together, these properties make the bilateral grid well-suited to streaming token compression.

However, generalizing the bilateral grid to visual token compression is non-trivial: the grid was originally designed for one-dimensional signals (i.e., 1D pixel intensity), whereas visual features are high-dimensional, causing the number of grid cells to explode and the memory cost to become intractable (Section 3.2). To address this, we provide a theoretical analysis based on the Johnson–Lindenstrauss lemma (Dasgupta & Gupta, 2003), showing that the semantic structure among visual tokens is preserved with high probability even after projecting their features into a low-dimensional guidance space. This enables splatting on a tractable grid, allowing BGStream to process streaming frames efficiently in both time and memory. In addition, following prior work (Di et al., 2025), we concatenate a FIFO buffer of recent frames with the grid tokens to retain high-fidelity perceptual evidence near the query timestamp.

We validate BGStream on two streaming video understanding benchmarks, StreamingBench (Lin et al., 2024) and OVO-Bench (Niu et al., 2025), where it significantly outperforms visual token compression baselines. We further show that BGStream is compatible with KV cache optimization methods, facilitating efficient streaming video understanding in real-world deployments.

Our contributions are summarized as follows: **(1)** We propose BGStream, a training-free method that splats visual tokens onto a bilateral grid, enabling on-the-fly compression without a temporal boundary prior or the full video. **(2)** We provide a theoretical analysis grounded in the Johnson–Lindenstrauss lemma, showing that the semantic structure is preserved with high probability within a tractable grid. **(3)** On two streaming video understanding benchmarks (StreamingBench (Lin et al., 2024) and OVO-Bench (Niu et al., 2025)), BGStream significantly outperforms prior visual token compression methods while remaining compatible with KV cache optimization.

2 Related Works

Grid Representation. The bilateral filter (Tomasi & Manduchi, 1998) smooths images while preserving edges, but its direct implementation is computationally expensive. To enable real-time performance, the bilateral grid (Paris & Durand, 2006) reformulates bilateral filtering as approximate Gaussian filtering in a lifted spatial–range space. This formulation has inspired several efficient variants, including Gaussian KD-trees (Adams et al., 2009) and adaptive manifolds (Gastal & Oliveira, 2012), and has been extended beyond filtering to stereo matching (Xu et al., 2021) and image registration (Zhang et al., 2024a). Unlike prior uses that remain limited to low-dimensional signals, we generalize the grid to high-dimensional visual tokens for streaming compression, a non-trivial extension that we make tractable via a theoretical analysis (Section 4.1.1).

Efficient Video MLLMs. Two approaches reduce the streaming cost of Video MLLMs: KV cache optimization, which curbs KV-cache memory growth inside the LLM (Di et al., 2025; Kim et al., 2025), and visual token compression, which reduces the number of visual tokens before they enter the LLM (Shao et al., 2025). The latter has mostly been studied offline, where the full video is available; adapting it to streaming is largely underexplored. BGStream addresses this with a bilateral grid that preserves semantic structure even under aggressive compression and without the full video, while remaining complementary to KV cache optimization. See Appendix A for further details.

[1]The retention ratio refers to the fraction of the full set of visual tokens retained after compression.

3 PRELIMINARY

3.1 STREAMING INFERENCE OF VIDEO MLLMS

A Video MLLM consists of a visual encoder, an adaptor, and an LLM. For an incoming frame at time t, the visual encoder produces visual features $\tilde{\mathbf{V}}_t \in \mathbb{R}^{n_v \times d_v}$, which the adaptor projects into visual tokens $\mathbf{V}_t \in \mathbb{R}^{n_v \times d}$ in the LLM's embedding space, where n_v is the number of visual tokens per frame and d_v, d are the embedding dimensions of the visual encoder and the LLM, respectively. In the streaming setting, frames arrive continuously, and the LLM can be queried at any timestamp t_q using only the frames observed up to that point. Formally, the visual context fed to the LLM at t_q is

$$\mathbf{V}_{1:t_q} = [\,\mathbf{V}_1;\, \mathbf{V}_2;\, \ldots;\, \mathbf{V}_{t_q}\,] \in \mathbb{R}^{(t_q \cdot n_v) \times d}. \tag{1}$$

Since $t_q \cdot n_v$ grows linearly with stream length and quickly exceeds the LLM's context window, the goal of streaming visual token compression is to construct a compact surrogate $\widehat{\mathbf{V}}_{1:t_q}$ whose size is much smaller than $t_q \cdot n_v$, while preserving the information needed to answer the query at t_q.

3.2 BILATERAL GRID

The bilateral grid (Paris & Durand, 2006) is a structured representation originally developed for edge-preserving image filtering. Its central idea is to lift each pixel into a higher-dimensional space by augmenting its 2D spatial coordinate with an additional axis that encodes pixel intensity, so that pixels close in this lifted space are similar in both spatial location and intensity. Formally, given an image $I : \Omega \to \mathbb{R}$ defined on a 2D spatial domain Ω, each pixel $p = (x, y)$ is lifted into

$$\Phi(p) = \big(x,\ y,\ I(x,y)\big) \in \mathbb{R}^2 \times \mathbb{R}, \tag{2}$$

where the third axis is referred to as the range domain. This lifted space is then discretized into a regular 3D grid Γ with spatial sampling rate s_s and range sampling rate s_r[2], yielding cells indexed by

$$\mathbf{u}(p) = \big(\lfloor x/s_s \rfloor,\ \lfloor y/s_s \rfloor,\ \lfloor I(x,y)/s_r \rfloor\big) \in \mathbb{Z}^3. \tag{3}$$

Here, the spatial coordinates (x, y) determine *where* each pixel lands, while the intensity $I(x, y)$—which we refer to as the key—routes pixels with similar intensity into the same cell.

Splatting. Pixels are deposited into the grid through an operation called splatting, which additively accumulates each pixel into its assigned cell. Following homogeneous coordinates (Chen et al., 2007), each cell $\Gamma[\mathbf{u}(p)]$ stores a pair $(sum,\ c)$: a running sum of intensities and a count c of deposited pixels, updated as

$$\Gamma[\mathbf{u}(p)] \leftarrow \Gamma[\mathbf{u}(p)] + \big(I(p),\ 1\big). \tag{4}$$

Through this additive form, multiple pixels can be incorporated into the same cell incrementally—pixel by pixel—without revisiting previously splatted ones. After splatting, the per-cell average $\bar{I}_{\mathbf{u}} = \Gamma[\mathbf{u}]_{sum}/\Gamma[\mathbf{u}]_c$ summarizes the intensities deposited in each cell into a single representative value. Combined with key-based routing, this aggregation makes the bilateral grid a compact, semantic structure-preserving surrogate of the image even under coarse discretization (Chen et al., 2016).

Challenge of generalizing the range domain to visual features. Discretizing a d_v-dimensional visual feature instead of 1D intensity makes the number of grid cells scale as $\lceil 1/s_r \rceil^{d_v}$, growing exponentially with d_v. For typical visual features ($d_v \geq 768$), even a coarse s_r yields a cell count far exceeding any feasible memory budget. Resolving this blow-up while preserving the semantic structure is the central technical challenge that BGStream addresses (Section 4.1).

4 PROPOSED METHOD: BGSTREAM

We present BGStream, a training-free visual token compression method (Figure 2). First, to address the challenge in Section 3.2, we project visual features into a low-dimensional guidance space, showing via the Johnson–Lindenstrauss (JL) lemma (Dasgupta & Gupta, 2003) that semantic structure is preserved (Section 4.1). Second, we splat each visual token onto a compact bilateral grid using its guidance key, aggregating tokens that share a cell (Section 4.2). Third, we maintain a short FIFO buffer of recent frames alongside the grid tokens to retain high-fidelity evidence near the query timestamp (Section 4.3).

[2] $s_s \geq 1$ is the spatial bin width; each cell covers an $s_s \times s_s$ pixel region. $s_r \in (0, 1]$ is the range bin width over normalized intensities $[0, 1]$, yielding $\lceil 1/s_r \rceil$ bins.

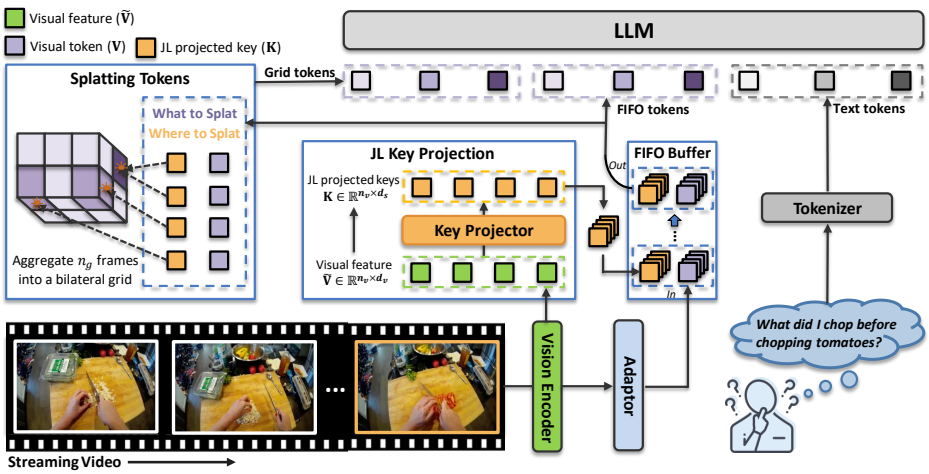


Figure 2: Overall framework of BGStream. Each incoming frame's visual features ($\tilde{\mathbf{V}}$) are projected to low-dimensional keys $\mathbf{K}$ via the Johnson–Lindenstrauss (JL) projection (*where to splat*), and the corresponding visual tokens $\mathbf{V}$ (*what to splat*) are splatted into a bilateral grid using these keys; tokens sharing a cell are merged into a single grid token, yielding a compact set that summarizes n_g frames. A FIFO buffer additionally retains tokens from the most recent n_r frames. The grid tokens, the FIFO tokens, and the tokenized query are fed to the LLM to generate the response.

4.1 Semantic Structure-Preserving Key Projection

Recall from Section 3.2 that bilateral grid splatting requires, for each token, a *key* that determines which cell it lands in, and that directly using the d_v-dimensional visual feature as the key is intractable. To resolve this, we leverage the JL lemma, which guarantees that a Gaussian random projection preserves the pairwise Euclidean distances of a finite point set with high probability—a property we exploit to preserve the semantic structure of visual tokens within a low-dimensional space. However, the classical JL bound requires the projection dimension d_s to scale logarithmically with the number of feature vectors N, which remains prohibitive at streaming scale. Our analysis therefore progressively relaxes this strict pairwise distance preservation to match what bilateral grid splatting actually requires, potentially allowing a substantially smaller d_s. Among JL-compatible projections, we adopt a Gaussian projection matrix for its analytical tractability. Specifically, given a visual feature matrix $\tilde{\mathbf{F}} \in \mathbb{R}^{N \times d_v}$[3], we obtain its keys via

$$\tilde{\mathbf{K}} = \tilde{\mathbf{F}}\mathbf{P}, \quad \mathbf{P} = \frac{1}{\sqrt{d_s}}\mathbf{G}, \quad \mathbf{G}_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1), \tag{5}$$

where $\mathbf{P} \in \mathbb{R}^{d_v \times d_s}$ is the Gaussian random projection matrix.

4.1.1 Theoretical Analysis

We adapt the JL guarantee to BGStream through three stages that progressively relax the geometric condition on the projection, potentially allowing smaller admissible values of d_s. See Appendix B for full proofs.

Stage (i): Strict pairwise isometry (Appendix B.1). We first analyze the strictest geometry condition—preserving all pairwise Euclidean distances. Adapting the classical JL inequalities to the Gaussian projector $\mathbf{P}$ in Eq. 5, we obtain a baseline bound (*Theorem* B.3): for any distortion $\varepsilon \in (0,1)$ and failure probability $\delta \in (0,1]$,

$$d_s > \frac{2\ln N - \ln \delta}{C(\varepsilon)}, \qquad C(\varepsilon) = \frac{\varepsilon^2}{4} - \frac{\varepsilon^3}{6}, \tag{6}$$

guarantees $(1-\varepsilon)\|\mathbf{f}_i - \mathbf{f}_j\|^2 \le \|(\mathbf{f}_i - \mathbf{f}_j)\mathbf{P}\|^2 \le (1+\varepsilon)\|\mathbf{f}_i - \mathbf{f}_j\|^2$ for all pairs with probability at least $1-\delta$. However, the union-bound penalty $\ln N^2$ over all $\binom{N}{2}$ pairs remains the fundamental bottleneck at streaming scale.

[3] N denotes the number of feature vectors fed jointly to the projector, which we keep abstract so that the analysis below applies to any N; the concrete choice ($N = n_g \cdot n_v$) is given in Section 4.2. We project visual features rather than visual tokens for computational efficiency, since $d_v < d$.

Stage (ii): Grid-cell semantic purity (Appendix B.2). We therefore relax the objective to SEMANTIC PURITY: each non-empty grid cell should contain features from a single semantic cluster. This suffices for grid-based merging, since intra-class splitting across cells is benign while inter-class collisions are destructive. Under the stated clusterability and projection assumptions[4], *Theorem* B.6 (and *Remark* B.7) guarantees semantic purity for all non-empty cells with probability at least $1 - \delta$. The micro-level contribution to the failure probability is bounded by $N\mathbb{P}_{\text{micro}}$, where

$$\mathbb{P}_{\text{micro}} \leq \frac{1}{\sqrt{\pi d_s}} \frac{1}{\varepsilon_p + 2/d_s} \exp\Big(- d_s I(1+\varepsilon_p) - \tfrac{1}{6d_s+1} \Big), \tag{7}$$

Here the geometry-dependent tolerance ε_p replaces the global ε of Stage (i). This relaxes the $\ln N^2$ penalty to a cluster-aware bound, substantially loosening the constraint on d_s.

Stage (iii): Aggregate semantic integrity (Appendix B.3). We further relax cell purity to aggregate robustness in expectation. Under the mild assumption inherited from Stage(ii), each cell is assigned a reference label based on the safety-region geometry. After splatting, select a cell J uniformly and independently of the projection from all L fixed grid indices. Let S_J and F_J denote its signal and noise counts. Theorem B.8 yields

$$\text{SNR}_{\text{unif}} := \frac{\mathbb{E}[S_J]}{\mathbb{E}[F_J]} = \frac{1 - p_{\text{fail}}}{p_{\text{fail}}} \geq \frac{1 - \mathbb{P}_{\text{micro}}}{\mathbb{P}_{\text{micro}}}, \tag{8}$$

where p_{fail} is the average source-token failure probability. Under assumptions on the value vectors, the theorem also bounds the expected representation error, with each cell weighted by its token count. Figure 4 shows that favorable geometry can yield a high SNR_{unif} even at small d_s, like $d_s = 4$ or 5.

Together, these analyses relate projection distortion, cell purity, and aggregate representation error under explicit geometric assumptions, providing a theoretical framework for interpreting the empirical dimension trade-off.

4.2 Splatting Tokens onto a Bilateral Grid

Building on the keys $\tilde{\mathbf{K}}$ derived in Section 4.1.1, we apply min-max normalization $\mathbf{K} = \text{minmax}(\tilde{\mathbf{K}})$ so that the keys lie in $[0,1]^{d_s}$, and then generalize the splatting mechanism of Section 3.2 from a 1D intensity range to the d_s-dimensional guidance key (detailed below). For the i-th visual token, we denote its key and value as $(\mathbf{k}_i, \mathbf{v}_i)$, determining *where* and *what* to splat, respectively. We use the visual token $\mathbf{v}_i \in \mathbb{R}^d$ itself as the value since it is ultimately fed to the LLM, while the key $\mathbf{k}_i \in \mathbb{R}^{d_s}$ only indicates its placement.

Grid construction. We lift each visual token into a $(2 + d_s)$-dimensional space spanned by its 2D patch coordinate (x_i, y_i)[5] and its guidance key $\mathbf{k}_i$, replacing the scalar intensity axis of Eq. 2. Discretizing this space with spatial sampling rate s_s and range sampling rate s_r yields a grid Γ with $\lceil 1/s_r \rceil^{d_s}$ range cells per spatial location.

Splatting and aggregation. Following the homogeneous-coordinate convention of Section 3.2, each cell of Γ stores a pair $(sum, c) \in \mathbb{R}^d \times \mathbb{Z}_{\geq 0}$, the d-dimensional analogue of the scalar (sum, c) pair used there. To deposit the i-th token, we first compute its splat index by coordinate-wise quantization of its spatial location and guidance key:

$$\mathbf{u}(x_i, y_i, \mathbf{k}_i) = \big(\lfloor x_i/s_s \rfloor, \lfloor y_i/s_s \rfloor, \lfloor \mathbf{k}_{i,1}/s_r \rfloor, \ldots, \lfloor \mathbf{k}_{i,d_s}/s_r \rfloor \big), \tag{9}$$

generalizing Eq. 3 by replacing the scalar intensity key with the d_s-dimensional guidance key. Following Eq. 4, the token then contributes its value to the assigned cell with a unit count,

$$\Gamma\big[\mathbf{u}(x_i, y_i, \mathbf{k}_i)\big] \;\leftarrow\; \Gamma\big[\mathbf{u}(x_i, y_i, \mathbf{k}_i)\big] + (\mathbf{v}_i, 1), \tag{10}$$

and the per-cell average summarizes each non-empty cell into a single representative token,

$$\bar{\mathbf{v}}_{\mathbf{u}} = \frac{\Gamma[\mathbf{u}]_{sum}}{\Gamma[\mathbf{u}]_c} \in \mathbb{R}^d, \tag{11}$$

[4]We assume K disjoint semantic clusters with maximum intra-cluster radius R and minimum inter-cluster separation Δ. The threshold τ_{grid} is chosen before projection to bound the realized grid ambiguity diameter $s_r \|\mathbf{W}(\mathbf{P})\|_2$ with high probability. For a prescribed centroid-contraction tolerance ε_g, the allowable point-expansion tolerance is $\varepsilon_p(\varepsilon_g) = ((1-\varepsilon_g)\Delta - \tau_{\text{grid}})/(2R) - 1 > 0$; full conditions are given in Appendix B.2.

[5]Unlike the pixel coordinate in Section 3.2, (x_i, y_i) here denotes the patch coordinate within a frame.

the multi-token, d-dimensional analogue of the per-cell intensity $\bar{I}_{\mathbf{u}}$ in Section 3.2.

Incorporating streaming frames. By design, our splatting mechanism is inherently *streaming-friendly*: each token is deposited into a single grid cell in one pass (Eq. 10), and representative tokens are computed *lazily* via Eq. 11 only when queried by the LLM. Frames can thus be splatted as they arrive, without revisiting past frames. To keep each grid bounded, we segment the stream into temporal groups of n_g consecutive frames ($N = n_g \cdot n_v$), building one grid per group with a shared projection matrix $\mathbf{P}$. Semantically similar tokens—from the same or neighboring frames—land in the same cell and are merged, eliminating temporal redundancy as a natural by-product, unlike prior approaches that use dedicated modules (Shao et al., 2025). Flattening the non-empty cells yields a compact set of grid tokens $\{\bar{\mathbf{v}}_{\mathbf{u}} : \Gamma[\mathbf{u}]_c > 0\} \subset \mathbb{R}^d$ summarizing the n_g frames, after which a new grid is constructed for the next group.

4.3 Preserving Recent Perceptual Evidence via a FIFO Buffer

While splatting compresses past frames into a compact set of grid tokens, it trades fine-grained perceptual detail for compactness, particularly in frames near the query timestamp t_q—precisely the frames likely to carry query-relevant evidence. Following prior streaming work (Di et al., 2025), we therefore maintain a FIFO buffer of the most recent n_r frames to supply the LLM with high-fidelity evidence near t_q. Under tight token budgets, we retain only the top-$\rho\%$ tokens per frame based on visual-encoder attention scores; see Appendix C.1 for details. See Appendix D.9 for analysis of the FIFO buffer size n_r.

4.4 Streaming Inference and Token Budgeting

When a user query arrives at timestamp t_q, the visual context $\widehat{\mathbf{V}}_{1:t_q}$ fed to the LLM is formed by concatenating the grid tokens accumulated up to t_q with the tokens in the FIFO buffer. Relative to the original $n_v \cdot t_q$ visual tokens, the retention ratio is

$$R = \frac{\underbrace{\lceil (t_q - n_r)/n_g \rceil \cdot \lceil 1/s_r \rceil^{d_s}}_{\text{Grid tokens}} + \underbrace{\rho \cdot n_v \cdot n_r}_{\text{FIFO tokens}}}{n_v \cdot t_q}, \tag{12}$$

where the grid term[6] counts only non-empty cells and grows linearly with t_q, while the FIFO term remains constant. When the total context $|\widehat{\mathbf{V}}_{1:t_q}|$ exceeds a target token budget B (e.g., the LLM's context window), we apply a lightweight cross-grid merging step that consolidates pairs of adjacent grids—coarsening older grids first while keeping recent ones fine-grained—until the context fits within B (see Appendix C.2).

5 Experiments

5.1 Experimental Setting

Datasets. We evaluate BGStream on two streaming benchmarks, StreamingBench (Lin et al., 2024) and OVO-Bench (Niu et al., 2025), with videos sampled at 0.5 FPS following prior work (Wang et al., 2025b; Di et al., 2025). To evaluate generality beyond streaming, we further evaluate on three offline benchmarks: EgoSchema (Mangalam et al., 2023), LongVideoBench (Wu et al., 2024), and MLVU (Zhou et al., 2025). Further dataset details and offline results are in Appendices D.1 and D.8.

Baselines. We use LLaVA-OneVision-7B (Li et al., 2024) as the backbone and compare BGStream against visual token compression baselines of two types: spatial (S), which reduces redundancy within each frame—ToMe (Bolya et al., 2022), VisionZip (Yang et al., 2025a), FastVid (Shen et al., 2025)—and spatial-temporal (S+T), which additionally removes redundancy across frames—PruneVid (Huang et al., 2025), Holitom (Shao et al., 2025), FlashVid (Fan et al., 2026). We also include StreamingTOM (Chen et al., 2025), the state-of-the-art streaming S+T method, and, as a reference, ReKV (Di et al., 2025), an orthogonal KV-cache offloading approach that retains **all visual tokens**. For fair comparison, all baselines use a 5% retention ratio. We use $d_s = 4$ for BGStream

[6]We set the spatial sampling rate s_s so that $\lfloor H/s_s \rfloor = \lfloor W/s_s \rfloor = 1$ (no spatial partitioning; see Appendix D.3), hence the grid term has no spatial factor.

Table 1: Results on StreamingBench (Lin et al., 2024) dataset. ReKV (Di et al., 2025) feeds all visual tokens into the LLMs, followed by KV cache optimization. The best results are shown in bold. "T" denotes the compression type, where "S" and "S+T" indicate spatial and spatial-temporal compression, respectively.

T	Method	OP	CR	TR	ATP	EU	ACP	SU	CS	PR	CT	Overall	Latency (ms)
	KV Cache Optimization + Visual Token Compression												
	ReKV (Di et al., 2025) (ICLR'25)	77.24	80.47	70.09	80.13	68.55	64.77	61.38	78.55	68.52	38.30	69.84	13,160
	+ BGStream	76.42	72.66	61.99	79.17	71.07	70.17	68.29	85.80	62.04	27.66	68.08	4,179
	Visual Token Compression												
S	ToMe (Bolya et al., 2022) (ICLR'23)	69.92	74.22	54.83	67.95	61.64	58.24	57.32	75.08	60.19	24.47	61.36	381
S	VisionZip (Yang et al., 2025a) (CVPR'25)	66.67	71.88	55.45	60.90	64.78	57.39	56.10	67.51	52.78	19.68	58.28	809
S	FastVid (Shen et al., 2025) (NeurIPS'25)	69.38	76.56	56.07	68.59	66.04	58.81	61.38	70.98	60.19	23.94	61.84	813
S+T	PruneVid (Huang et al., 2025) (ACL'25)	73.17	81.25	59.19	71.15	66.67	63.64	59.76	78.86	64.81	**33.51**	65.84	885
S+T	Holitom (Shao et al., 2025) (NeurIPS'25)	70.19	78.12	57.63	70.51	66.04	60.80	59.35	76.34	65.74	27.13	63.72	1,182
S+T	FlashVid (Fan et al., 2026) (ICLR'26)	71.82	77.34	57.94	70.51	64.15	60.23	57.32	75.08	63.89	29.79	63.52	891
S+T	StreamingTom (Chen et al., 2025) (CVPR'26)	68.02	**81.25**	57.63	64.10	63.52	57.95	56.50	70.98	66.67	31.38	61.60	1,161
S+T	BGStream	**77.24**	74.22	**64.17**	**75.00**	**72.96**	**65.62**	**60.57**	**86.44**	**69.44**	26.60	**68.60**	815

Table 2: Results on OVO-Bench (Niu et al., 2025) dataset.

T	Method	Real-Time Visual Perception							Backward Tracing				Overall	Latency (ms)
		OCR	ACR	ATR	STU	FPD	OJR	Avg.	EPM	ASI	HLD	Avg.		
	KV Cache Optimization + Visual Token Compression													
	ReKV (Di et al., 2025) (ICLR'25)	72.48	60.55	70.91	43.82	73.27	60.33	61.97	58.59	57.43	29.03	49.60	56.63	14,400
	+ BGStream	60.40	61.47	78.18	46.07	67.33	67.39	62.21	51.85	52.70	25.81	44.37	54.51	6,156
	Visual Token Compression													
S	ToMe (Bolya et al., 2022) (ICLR'23)	44.30	50.46	65.45	37.64	69.31	52.72	51.38	49.16	50.00	**29.57**	43.58	48.02	487
S	VisionZip (Yang et al., 2025a) (CVPR'25)	38.26	42.20	58.18	38.20	63.37	45.65	46.09	46.46	45.95	19.35	38.35	42.75	1,024
S	FastVid (Shen et al., 2025) (NeurIPS'25)	44.30	49.54	61.82	39.89	68.32	55.98	51.87	46.46	47.97	17.20	38.19	45.96	1,026
S+T	PruneVid (Huang et al., 2025) (ACL'25)	53.02	55.96	**70.00**	39.89	69.31	53.26	54.87	50.51	47.97	17.20	40.10	48.50	1,109
S+T	Holitom (Shao et al., 2025) (NeurIPS'25)	49.66	53.21	68.18	37.64	67.33	52.72	52.83	49.16	51.35	18.82	40.73	47.61	1,473
S+T	FlashVid (Fan et al., 2026) (ICLR'26)	54.36	50.46	65.45	40.45	67.33	55.43	54.15	51.18	52.03	18.82	41.84	48.84	1,141
S+T	StreamingTOM (Chen et al., 2025) (CVPR'26)	50.34	48.62	60.00	37.64	62.38	51.09	50.30	46.13	54.05	28.49	42.79	47.06	1,425
S+T	BGStream	**65.10**	**62.39**	63.64	**44.38**	**70.30**	**67.39**	**61.25**	**53.87**	**53.38**	25.81	**45.80**	**54.58**	1,037

to match this budget, even though a smaller d_s further lowers its retention ratio while improving accuracy (Figure 3); we thus evaluate BGStream under a deliberately conservative, less compressive setting. See Appendix D.2 for additional baseline details.

Implementation Details. We evaluate BGStream on two Video MLLM backbones with distinct visual token granularities: LLaVA-OneVision-7B (Li et al., 2024) (n_v=196) and InternVL3.5-8B (Wang et al., 2025a) (n_v=256). All experiments are conducted on a single NVIDIA H200 GPU. See Appendix D.3 for additional implementation details.

5.2 Quantitative Results

We compare BGStream against baselines and report per-sample inference latency for efficiency.

Results on StreamingBench. Table 1 reports the results, from which we make three key observations. **1) BGStream surpasses all baselines in overall accuracy.** While spatial-temporal (S+T) methods generally outperform spatial-only (S) methods by exploiting temporal redundancy, BGStream surpasses all of them, demonstrating the effectiveness of preserving the semantic structure via the bilateral grid. **2) BGStream achieves spatial-level latency.** Prior S+T methods rely on pairwise similarity computation across tokens and frames, the major efficiency bottleneck. Although BGStream also exploits temporal redundancy (Section 4.2), it processes n_g frames jointly via a single JL projection followed by grid-based aggregation, eliminating pairwise computation and thus matching spatial-only latency **3) BGStream composes with KV cache optimization.** Applied on top of ReKV, BGStream substantially reduces latency with only marginal accuracy drop, demonstrating complementarity with existing KV cache techniques and enabling joint memory–compute optimization for real-world deployment. For additional complementarity results with KV cache optimization, please refer to Appendix D.11.

Results on OVO-Bench. Table 2 shows the same trends: BGStream surpasses all baselines in overall accuracy with spatial-level latency, and remains complementary to KV cache optimization. Beyond real-time perception, BGStream also achieves the best score on Backward Tracing, which evaluates recall of past events. This simultaneous gain on both perception and recall indicates that our grid representation effectively captures the semantic structure of streaming videos; see Appendix D.4 for additional controlled experiments.

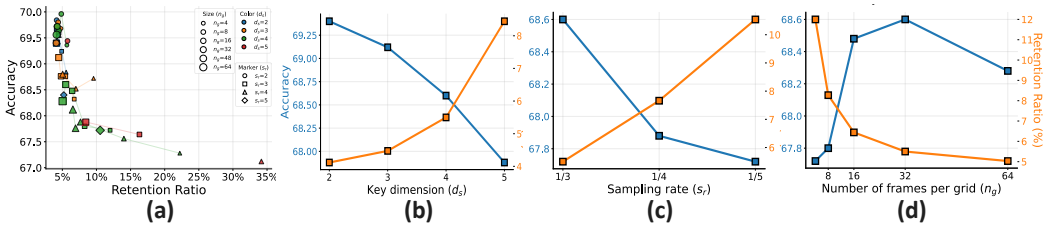


Figure 3: Factors affecting grid-cell merging on StreamingBench. **(a)** Accuracy vs. retention ratio overall all configurations (color: d_s, shape: s_r, size: n_g). **(b–d)** Each factor's effect on accuracy (blue, left) and retention ratio (orange, right): **(b)** JL projection dimension d_s, **(c)** range sampling rate s_r, and **(d)** number of frames per grid n_g.

5.3 In-Depth Analysis

We study three factors affecting grid-cell merging: the JL projection dimension d_s, the range sampling rate s_r, and the temporal group size n_g. Together, they determine which tokens share a cell, shaping accuracy and the retention ratio (Eq. 12). Figure 3(a) shows that BGStream maintains strong accuracy across a wide range of configurations—consistently surpassing the baselines in Table 1—with a clear knee around the 5% retention regime.

Grid resolution (d_s and s_r). Since the total grid cell count scales as $\lceil 1/s_r \rceil^{d_s}$, d_s and s_r jointly control the grid resolution. Figure 3(b,c) shows that either increasing d_s or decreasing s_r yields a finer grid, dispersing tokens across more cells and thus leaving fewer tokens per cell, which weakens merging effect—raising the retention ratio and degrading accuracy. Conversely, a coarser grid enforces stronger merging, and our theoretical analysis (Section 4.1.1) guarantees that the semantic structure needed for splatting is preserved even at very low d_s under mild assumptions. Empirically, d_s as small as 3 already yields a favorable accuracy–retention trade-off—in fact, lowering d_s both reduces the retention ratio and improves accuracy, which would favor BGStream. We nonetheless adopt $d_s = 4$ so that BGStream's retention ratio matches the $\sim$5% budget of the baselines, ensuring a fair—if conservative—comparison.

Temporal group size n_g. n_g determines how many consecutive frames share a single grid, thereby exploiting their temporal redundancy. Figure 3(d) shows that accuracy improves substantially as n_g grows, peaks around $n_g = 32$, and declines slightly beyond that—we attribute this drop to over-aggressive merging across temporally distant frames whose semantic content has drifted. Meanwhile, the retention ratio decreases monotonically with n_g, as more frames are summarized by a single grid. A moderate n_g thus strikes the best balance between temporal redundancy reduction and semantic fidelity.

For the ablation studies of the FIFO buffer and alternative projection methods, please refer to Appendix D.12.

5.4 Experiments with Different Backbones

To verify that BGStream generalizes beyond a specific backbone, we evaluate it on InternVL3.5-8B (Wang et al., 2025a), which differs from LLaVA-OneVision in both the visual encoder and the language model. As shown in Table 3, we observe that BGStream consistently surpasses all visual token compression baselines on both StreamingBench and OVO-Bench by substantial margins and remains complementary to KV cache optimization. This confirms its versatility across diverse Video MLLM backbones. Regarding the experimental results of per-category, please see Appendix D.5. Regarding the results with another backbone (LLaVA-Video (Zhang et al., 2024c)), please see Appendix D.6.

Table 3: Performance comparison using InternVL3.5 (Wang et al., 2025a) as the backbone.

T	Method	StreamingBench	OVO-Bench
	KV Cache Optimization + Visual Token Compression		
	ReKV (Di et al., 2025)	72.80	56.77
	+ BGStream	68.80	54.10
	Visual Token Compression		
S	ToMe (Bolya et al., 2022)	63.64	47.33
S	VisionZip (Yang et al., 2025a)	61.28	44.25
S	FastVid (Shen et al., 2025)	63.36	46.17
S+T	PruneVid (Huang et al., 2025)	65.92	48.29
S+T	Holitom (Shao et al., 2025)	65.00	47.81
S+T	FlashVid (Fan et al., 2026)	66.20	47.33
S+T	StreamingTOM (Chen et al., 2025)	64.48	48.63
	BGStream	**70.08**	**54.45**

5.5 Different Sampling FPS

To study how the sampling FPS affects accuracy and latency, we evaluate on StreamingBench with FPS from 0.2 to 1.0, using ReKV as a reference that retains all tokens. Table 4 highlights two observations. **1) Higher FPS improves accuracy at the cost of latency.** As FPS increases, latency rises across all methods, while accuracy generally improves, indicating that higher FPS captures finer temporal dynamics at the cost of heavier frame processing. **2) BGStream scales favorably with FPS.** Owing to its lightweight splatting mechanism, BGStream keeps latency low as FPS increases while steadily improving accuracy, even surpassing ReKV at FPS 1. We attribute this to ReKV's overhead from retaining all tokens as FPS grows, whereas BGStream suppresses redundancy while preserving the semantic structure—sustaining accuracy without latency blowup as FPS scales.

Table 4: Performance and latency across different FPS. ReKV (all visual tokens) is shown as a reference.

Sample FPS	Method	Accuracy (%)	Latency (ms)
0.2	ReKV	**68.8**0	5,481
	StreamingTom	58.92	552
	BGStream	66.64	**346**
0.5	ReKV	**69.84**	13,160
	StreamingTom	61.60	1,162
	BGStream	68.60	**815**
1	ReKV	67.84	22,052
	StreamingTom	62.08	2,144
	BGStream	**69.96**	**1,567**

5.6 Distortion and Semantic Purity

To empirically validate our theoretical analysis on the preservation of semantic structure, we measure the pairwise distortion ϵ and semantic purity of the compressed tokens (metric definitions in Appendix D.7). Table 5 shows that, at an aggressive retention ratio of 5%, baselines exhibit high distortion and low purity, indicating that conventional compression degrades the semantic structure under tight token budgets. In contrast, BGStream achieves the lowest distortion and highest purity at the same budget, surpassing every baseline by a large margin. Remarkably, BGStream at 5% matches baselines at 10% retention, using only a fraction of their token budget. These results provide complementary empirical evidence for our theoretical claim that BGStream preserves the semantic structure even under aggressive compression. Beyond this, we further analyze the number of non-empty cells per grid, both theoretically and empirically, in Appendix B.4.

Table 5: Distortion and semantic purity across retention ratios; lower distortion and higher purity indicate better preservation of the semantic structure.

	Distortion $\epsilon \downarrow$		Purity $\uparrow$	
Method	**5%**	**10%**	**5%**	**10%**
ToMe	0.204	0.174	0.713	0.757
StreamingTom	0.314	0.261	0.652	0.721
PruneVid	0.178	**0.148**	0.780	0.816
Holitom	0.180	**0.148**	0.786	0.822
BGStream	**0.148**	0.152	**0.846**	**0.839**

6 Conclusion

We revisited visual token compression under a setting it was not designed for. Two obstacles separate existing methods from streaming video understanding: the unbounded frame stream forces aggressive retention ratios, under which prior methods collapse; and these methods presuppose a temporal boundary prior available only from the full video, unrealistic in streaming. BGStream overcomes both via the bilateral grid, whose splatting incorporates frames without the full video and whose aggregation stays effective under aggressive retention. The challenge this raised—that handling high-dimensional visual features on a bilateral grid is intractable—was resolved not by approximation but by argument: grounding the design in the Johnson–Lindenstrauss lemma, we showed that the semantic structure among visual tokens is preserved within a tractable grid, and experiments support semantic structure preservation in practice.

7 Limitations

BGStream comes from a deliberate trade-off: a coarse grid merges semantically close tokens aggressively, which is what enables its strong compression but also discards fine distinctions among them. This is visible on the Counting (CT) sub-task of StreamingBench (Table 1), where numerically distinct entities that are visually similar can fall into the same cell and be merged, blurring the count. Our occupancy analysis (Appendix B.4) makes this explicit: at the coarse grid we adopt, cells are nearly saturated, so tokens are merged precisely where the strongest compression occurs. Recovering such fine distinctions without sacrificing compression—for instance, by adapting the grid resolution to the local semantic density rather than fixing it globally—is a natural next step. Furthermore, extending BGStream beyond single-camera streaming to multi-view or audio-visual settings is another promising direction.

USE OF AI ASSISTANTS

Throughout this paper, we used AI assistants (e.g., Claude, GPT) for limited assistance with code completion and language polishing of the manuscript. All research ideas, experimental design, analysis, and conclusions are the authors' own. No AI assistant was used to generate experimental results or scientific claims.

ETHICS STATEMENT

To the best of our knowledge, this work complies with the ICLR Code of Ethics and poses no ethical concerns. Additionally, all datasets and pre-trained models used in our experiments are publicly available.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our results, we describe the experimental setup and implementation details in Section 5.1 and Appendix D.3, respectively.

REFERENCES

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

Andrew Adams, Natasha Gelfand, Jennifer Dolson, and Marc Levoy. Gaussian kd-trees for fast high-dimensional filtering. In *ACM SIGGRAPH 2009 papers*, pp. 1–12. Association for Computing Machinery, 2009.

Raghu Raj Bahadur and R Ranga Rao. On deviations of the sample mean. *The Annals of Mathematical Statistics*, 31(4):1015–1027, 1960. doi: 10.1214/aoms/1177705674.

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.

Jiawen Chen, Sylvain Paris, and Frédo Durand. Real-time edge-aware image processing with the bilateral grid. *ACM Transactions on Graphics (TOG)*, 26(3):103–es, 2007.

Jiawen Chen, Andrew Adams, Neal Wadhwa, and Samuel W Hasinoff. Bilateral guided upsampling. *ACM Transactions on Graphics (ToG)*, 35(6):1–8, 2016.

Xueyi Chen, Keda Tao, Kele Shao, and Huan Wang. Streamingtom: Streaming token compression for efficient video understanding. *arXiv preprint arXiv:2510.18269*, 2025.

Yilong Chen, Xiang Bai, Zhibin Wang, Chengyu Bai, Yuhan Dai, and Ming Lu. Streamkv: Streaming video question-answering with segment-based kv cache retrieval and compression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3120–3128, 2026. doi: 10.1609/aaai.v40i4.37305.

Janghoon Cho, Jungsoo Lee, Munawar Hayat, Kyuwoong Hwang, Fatih Porikli, and Sungha Choi. Floc: Facility location-based efficient visual token compression for long video understanding. *arXiv preprint arXiv:2511.00141*, 2025.

Sanjoy Dasgupta and Anupam Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003. doi: 10.1002/rsa.10073.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Tao Zhong, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, and Hao Jiang. Streaming video question-answering with in-context video kv-cache retrieval. *arXiv preprint arXiv:2503.00540*, 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.

Eduardo SL Gastal and Manuel M Oliveira. Adaptive manifolds for real-time high-dimensional filtering. *ACM Transactions on Graphics (TOG)*, 31(4):1–13, 2012.

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.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pp. 604–613, 1998.

Minsoo Kim, Kyuhong Shim, Jungwook Choi, and Simyung Chang. Infinipot-v: Memory-constrained kv cache compression for streaming video understanding. *arXiv preprint arXiv:2506.15745*, 2025.

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*, 2024.

Junming Lin, Zheng Fang, Chi Chen, Zihao Wan, Fuwen Luo, Peng Li, Yang Liu, and Maosong Sun. Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. *arXiv preprint arXiv:2411.03628*, 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.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*. Cambridge University Press, 2008. doi: 10.1017/CBO9780511809071.

Zhenyu Ning, Guangda Liu, Qihao Jin, Wenchao Ding, Minyi Guo, and Jieru Zhao. Livevlm: Efficient online video understanding via streaming-oriented kv cache and retrieval. *arXiv preprint arXiv:2505.15269v1*, 2025. URL https://arxiv.org/abs/2505.15269v1.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, et al. Ovo-bench: How far is your video-llms from real-world online video understanding? In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18902–18913, 2025.

Sylvain Paris and Frédo Durand. A fast approximation of the bilateral filter using a signal processing approach. In *European conference on computer vision*, pp. 568–580. Springer, 2006.

Jiayuan Rao, Haoning Wu, Hao Jiang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Towards universal soccer video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8384–8394, 2025.

Herbert Robbins. A remark on stirling's formula. *The American mathematical monthly*, 62(1):26–29, 1955. doi: 10.2307/2308012.

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, and Guiguang Ding. Fastvid: Dynamic density pruning for fast video large language models. *arXiv preprint arXiv:2503.11187*, 2025.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, et al. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*, 2024.

Yujiao Shen, Shulin Tian, Jingkang Yang, and Ziwei Liu. A simple baseline for streaming video understanding. *arXiv preprint arXiv:2604.02317*, 2026.

Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pp. 839–846. IEEE, 1998.

Pavan Kumar Anasosalu Vasu, Fartash Faghri, Chun-Liang Li, Cem Koc, Nate True, Albert Antony, Gokula Santhanam, James Gabriel, Peter Grasch, Oncel Tuzel, et al. Fastvlm: Efficient vision encoding for vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 19769–19780, 2025.

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48 of *Cambridge Series in Statistical and Probabilistic Mathematics*. Cambridge University Press, 2019.

Ao Wang, Fengyuan Sun, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. [cls] token tells everything needed for training-free efficient mllms. *arXiv preprint arXiv:2412.05819*, 2024.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025a.

Yiyu Wang, Xuyang Liu, Xiyan Gui, Xinying Lin, Boxue Yang, Chenfei Liao, Tailai Chen, and Linfeng Zhang. Accelerating streaming video large language models via hierarchical token compression. *arXiv preprint arXiv:2512.00891*, 2025b.

Zichen Wen, Yiyu Wang, Chenfei Liao, Boxue Yang, Junxian Li, Weifeng Liu, Haocong He, Bolong Feng, Xuyang Liu, Yuanhuiyi Lyu, et al. Ai for service: Proactive assistance with ai glasses. *arXiv preprint arXiv:2510.14359*, 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.

Bin Xu, Yuhua Xu, Xiaoli Yang, Wei Jia, and Yulan Guo. Bilateral grid learning for stereo matching networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12497–12506, 2021.

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, 2025a.

Yanlai Yang, Zhuokai Zhao, Satya Narayan Shukla, Aashu Singh, Shlok Kumar Mishra, Lizhu Zhang, and Mengye Ren. Streammem: Query-agnostic kv cache memory for streaming video understanding. *arXiv preprint arXiv:2508.15717*, 2025b.

Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, et al. Streamforest: Efficient online video understanding with persistent event memory. *arXiv preprint arXiv:2509.24871*, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Hang Zhang, Xiang Chen, Rongguang Wang, Renjiu Hu, Dongdong Liu, and Gaolei Li. Slicer networks. *arXiv preprint arXiv:2401.09833*, 2024a.

Haowei Zhang, Shudong Yang, Jinlan Fu, See-Kiong Ng, and Xipeng Qiu. Hermes: Kv cache as hierarchical memory for efficient streaming video understanding. *arXiv preprint arXiv:2601.14724*, 2026.

Qizhe Zhang, Aosong Cheng, Ming Lu, Zhiyong Zhuo, Minqi Wang, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. [cls] attention is all you need for training-free visual token pruning: Make vlm inference faster. *arXiv preprint arXiv:2412.01818v1*, 2024b. URL `https://arxiv.org/abs/2412.01818v1`.

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*, 2024c.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, et al. Mlvu: Benchmarking multi-task long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13691–13701, 2025.

Supplementary Material

- Structure-Preserving Visual Token Compression for Streaming Video Understanding via Bilateral Grid Splatting -

A Additional Related Work **14**

A.1 Efficient Video MLLMs 14

B Detailed Theoretical Analysis **14**

B.1 Strict Geometric Isometry: Sharpened Johnson-Lindenstrauss Bounds 14

B.2 Grid-Cell Semantic Purity: Preservation of Cluster Separation 21

B.3 Aggregate Semantic Integrity: Statistical Robustness of Merged Tokens 27

B.4 Expectation of Vacancy Rate of Bilateral Grids 30

C Details of Method **33**

C.1 Attention-based Token Selection in the FIFO Buffer 33

C.2 Cross-Grid Merging under a Token Budget 33

D Experiments **34**

D.1 Additional Details of Dataset 34

D.2 Additional Details of Baselines 34

D.3 Additional Implementation Details 34

D.4 Comparison under Per-Query Matched Retention Ratios 35

D.5 Detailed Performance with InternVL3.5 Backbone 35

D.6 Results with the LLaVA-Video Backbone 36

D.7 Distortion and Semantic Purity Metrics 37

D.8 Experiments on Offline Long Video Understanding Benchmarks 37

D.9 Effect of FIFO Buffer Size 38

D.10 Evaluation on Contextual Understanding and Forward Active Responding 38

D.11 Comparison with Recent KV-Cache Methods 39

D.12 Ablation Study 39

D.13 Normalization Scope in Streaming Setting 39

D.14 Analysis of Computational Cost 40

A Additional Related Work

A.1 Efficient Video MLLMs

Efficient streaming video understanding. Deploying Video MLLMs for streaming video understanding faces two primary computational bottlenecks: (i) the memory footprint of the KV cache, which grows unboundedly as frames continuously arrive, and (ii) the LLM's quadratic attention cost over an ever-expanding visual token sequence. The predominant line of streaming research has focused on the former, **optimizing the KV cache** inside the LLM. Specifically, ReKV (Di et al., 2025) offloads KV entries to CPU memory and retrieves relevant ones at query time. LiveVLM (Ning et al., 2025) and StreamMem (Yang et al., 2025b) design streaming-oriented KV cache structures that balance recency and relevance, while Hermes (Zhang et al., 2026) organizes the KV cache hierarchically as memory. StreamKV (Chen et al., 2026) performs segment-based KV cache retrieval and compression, and InfiniPot-V (Kim et al., 2025) compresses the KV cache under a fixed memory budget. While effective at managing memory, these methods do not fundamentally address the computational cost of processing incoming visual tokens, leaving the LLM's attention cost largely intact. To address this, a complementary direction **compresses visual tokens** before they enter the LLM, directly cutting attention cost. This direction has only recently been explored for streaming inference: StreamingTOM (Chen et al., 2025) performs spatial-temporal token merging tailored to streaming arrivals, and STC (Wang et al., 2025b) reduces computation within the visual encoder by leveraging adjacent frames.

Visual token compression for offline video understanding. Visual token compression has been more extensively studied in the offline setting, where the entire video is available at compression time. Existing methods generally fall into two categories. Spatial-only methods, such as ToMe (Bolya et al., 2022), VisionZip (Yang et al., 2025a), FastVid (Shen et al., 2025), and FastVLM (Vasu et al., 2025), reduce per-frame token redundancy independently. Spatial-temporal methods, including PruneVid (Huang et al., 2025), Holitom (Shao et al., 2025), FlashVid (Fan et al., 2026), and FLoC (Cho et al., 2025), additionally exploit cross-frame redundancy by dividing the video into segments and removing temporally redundant tokens within each segment. While effective at moderate retention ratios, these methods struggle in aggressive compression required by streaming inference. Moreover, spatial-temporal methods presuppose access to the full video at compression time—an assumption fundamentally violated in streaming, where future frames are unavailable. In contrast, BGStream can effectively preserves semantic structure even under aggressive retention—precisely what streaming inference demands—without relying on future frames, and remains complementary to KV cache optimization, jointly enabling memory and compute efficiency for real-world streaming deployment.

B Detailed Theoretical Analysis

In Sec. 4.1, we proposed the key projector to map high-dimensional visual features $\tilde{\mathbf{F}} \in \mathbb{R}^{N \times d_v}$ into a compact guidance space $\mathbb{R}^{N \times d_s}$. A fundamental challenge in this design is establishing a rigorous relationship between the projection dimension d_s and the preservation of semantic structure. Without theoretical guarantees, the dimension reduction risks collapsing distinct semantic concepts. Conversely, an overly large guidance dimension increases the potential grid size and can make compression less effective. In this section, we provide a comprehensive mathematical theory to support our design, deriving explicit probability bounds that guide the selection of a minimal yet effective d_s.

Our theoretical framework validates this projection mechanism through a progressive relaxation of geometric constraints, transitioning from ideal mathematical rigor to practical usage. Specifically, we move from strict pairwise distance preservation (Theorem B.3) and grid-cell semantic purity (Theorem B.6), to ultimately establishing aggregate semantic robustness on expected-count SNR(Theorem B.8).

B.1 Strict Geometric Isometry: Sharpened Johnson-Lindenstrauss Bounds

We use a consistent projection convention throughout this appendix, matching the key projector in Sec. 4.1. Let $d_v > d_s$ be positive integers, and define

$$\mathbf{G} \in \mathbb{R}^{d_v \times d_s}, \qquad G_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1), \qquad \mathbf{P} := \frac{1}{\sqrt{d_s}}\mathbf{G}.$$

Here, $\mathbf{G}$ always denotes the unscaled standard Gaussian matrix, whereas $\mathbf{P}$ denotes the normalized projector used to construct the guidance keys; in particular, $P_{ij} \sim \mathcal{N}(0, 1/d_s)$. The same draw of $\mathbf{G}$, and hence the same $\mathbf{P}$, is shared by all feature vectors.

For any fixed unit vector $\mathbf{x} \in \mathbb{R}^{d_v}$, define the auxiliary Gaussian vector $\mathbf{r} := \mathbf{x}\mathbf{G}$. Then

$$\|\mathbf{r}\|^2 \sim \chi^2_{d_s}, \qquad \mathbf{x}\mathbf{P} = \frac{\mathbf{r}}{\sqrt{d_s}}, \qquad \|\mathbf{x}\mathbf{P}\|^2 = \frac{1}{d_s}\|\mathbf{r}\|^2.$$

We first derive tail bounds for the unscaled squared norm $\|\mathbf{r}\|^2$ and then apply them to the normalized projector $\mathbf{P}$. Thus, for $\epsilon \in (0, 1)$, the squared-distance preservation event is

$$1 - \epsilon \le \|\mathbf{x}\mathbf{P}\|^2 \le 1 + \epsilon,$$

equivalently, $d_s(1 - \epsilon) \le \|\mathbf{r}\|^2 \le d_s(1 + \epsilon)$.

Early works, such as (Indyk & Motwani, 1998, Lem. 6), established foundational bounds for this problem. These inspired Lemma 2.2 of (Dasgupta & Gupta, 2003) to derive sharper estimates specifically for standard random orthogonal matrices without the $\mathcal{O}(d_s)$ term. As a first step, we generalize the result of (Dasgupta & Gupta, 2003, Lem. 2.2) all standard random Gaussian matrices using the Chernoff bound (Wainwright, 2019, Eq. (2.5)), establishing a rigorous baseline before introducing our sharpened refinement.

Lemma B.1. *Let $\mathbf{x} \in \mathbb{R}^{d_v}$ be a unit vector and $\mathbf{G} \in \mathbb{R}^{d_v \times d_s}$ be a standard Gaussian random matrix. Define the auxiliary Gaussian vector as $\mathbf{r} = \mathbf{x}\mathbf{G}$. For any $\beta > 1$, the squared norm $\|\mathbf{r}\|^2$ satisfies the following tail bounds:*

1) (Upper Tail) $\mathbb{P}[\|\mathbf{r}\|^2 \ge \beta d_s] \le e^{-\frac{d_s}{2}(\beta - (1 + \ln \beta))}$,

2) (Lower Tail) $\mathbb{P}[\|\mathbf{r}\|^2 \le d_s/\beta] \le e^{-\frac{d_s}{2}\left(\beta^{-1} - (1 - \ln \beta)\right)}$.

Proof. Let $\mathbf{r} = (r_1, \ldots, r_{d_s})$. Since $\mathbf{x}$ is a unit vector, each entry r_i is a linear combination of independent standard Gaussian variables with unit variance. Consequently, $r_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$. It follows that the squared norm $\|\mathbf{r}\|^2 = \sum_{i=1}^{d_s} r_i^2$ follows a Chi-square distribution with d_s degrees of freedom, denoted as $\chi^2_{d_s}$. As we know, the moment generating function (MGF) of this distribution is given

$$M_{\|\mathbf{r}\|^2}(t) = \mathbb{E}[e^{t\|\mathbf{r}\|^2}] = (1 - 2t)^{-d_s/2}, \quad t \in (-\infty, \frac{1}{2}).$$

To obtain the upper tail inequality, we apply Markov's inequality to the random variable $Z = e^{t\|\mathbf{r}\|^2}$ for any $t \in (0, \frac{1}{2})$

$$\mathbb{P}[\|\mathbf{r}\|^2 \ge \beta d_s] = \mathbb{P}\left[e^{t\|\mathbf{r}\|^2} \ge e^{t\beta d_s}\right] \le e^{-t\beta d_s} \cdot M_{\|\mathbf{r}\|^2}(t).$$

By Chernoff bound, we can try to minimize the bound function

$$f(t) = e^{-t\beta d_s} \cdot \mathbb{E}\left[e^{t\|\mathbf{r}\|^2}\right] = e^{-t\beta d_s}(1 - 2t)^{-d_s/2}$$

in the range $(0, \frac{1}{2})$. The derivative of the power of f is

$$\frac{d}{dt}\left(-t\beta d_s - \frac{d_s}{2}\ln(1 - 2t)\right) = -\beta d_s + \frac{d_s}{1 - 2t}.$$

Setting its derivative to zero, we obtain the optimal point $t = 1/2(1 - 1/\beta)$, which lives in $(0, 1/2)$. Furthermore, the second derivative is strictly positive, confirming that t^* corresponds to a minimum. Substitute $t = 1/2(1 - 1/\beta)$ in $f(t)$, we obtain

$$\begin{aligned}
\mathbb{P}[\|\mathbf{r}\|^2 \ge \beta d_s] &\le f(t)_{\min} \\
&= e^{-\frac{\beta d_s}{2}(1 - \frac{1}{\beta}) \cdot \left(\frac{1}{\beta}\right)^{-d_s/2}} \\
&= e^{-\frac{d_s}{2}(\beta - 1 - \ln \beta)}.
\end{aligned}$$

Similarly, for any $t < 0$, applying the Chernoff bound yields:

$$\begin{aligned}
\mathbb{P}[\|\mathbf{r}\|^2 \le d_s/\beta] &= \mathbb{P}\left[e^{t\|\mathbf{r}\|^2} \ge e^{td_s/\beta}\right] \\
&\le e^{-td_s/\beta}(1 - 2t)^{-d_s/2}.
\end{aligned}$$

By similar calculation for the derivative of the bound function, we find the minimal point $t = \frac{1}{2}(1-\beta)$, thereby obtaining the required bound

$$\begin{aligned} e^{-\frac{1}{\beta}td_s}(1-2t)^{-d_s/2} &= e^{-\frac{d_s}{2}(\frac{1}{\beta}-1)} \cdot \beta^{-k/2} \\ &= e^{-\frac{d_s}{2}(\beta^{-1}-1+\ln\beta)}. \end{aligned}$$

□

As a natural development of Lemma B.1, we adopt the Taylor approximation used in the proof of (Dasgupta & Gupta, 2003, Thm. 2.1) to obtain the next proposition.

Proposition B.2. *Following the notation in Lemma B.1, for any $\epsilon \in (0,1)$, the squared norm of the auxiliary Gaussian vector $\|\mathbf{r}\|^2$ satisfies the following simplified tail bounds:*

1) (Upper Tail) $\mathbb{P}[\|\mathbf{r}\|^2 \geq (1+\epsilon)d_s] \leq e^{-\frac{d_s}{2}\left(\epsilon^2/2-\epsilon^3/3\right)}$,

2) (Lower Tail) $\mathbb{P}[\|\mathbf{r}\|^2 \leq (1-\epsilon)d_s] \leq e^{-\frac{d_s}{4}\epsilon^2}$.

Proof. We derive these bounds by applying Taylor approximations.

1) Let $\beta = 1+\epsilon$. We replace the logarithm term in the upper tail bound in the Lemma B.1 by $\ln\beta = \ln(1+\epsilon) \leq \epsilon - \epsilon^2/2 + \epsilon^3/3$.

1) Let $\beta^{-1} = 1-\epsilon$. We replace the logarithm term in the lower tail bound in the Lemma B.1 by $-\ln\beta = \ln(1-\epsilon) \leq -\epsilon - \epsilon^2/2$.

□

Having established the tail bounds for a single projected vector in B.2, we now extend this result to preserve the geometry of an entire dataset. By applying these bounds to all pairwise difference vectors and utilizing Boole's inequality (union bound) to control the cumulative failure risk, we derive the following Euclidean distance-preserving theorem.

Theorem B.3. *Let $A = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathbb{R}^{d_v}$ be a finite set of N points. Let $\epsilon \in (0,1)$ be a distortion parameter and $\delta \in (0,1]$ be a failure probability. If the projection dimension d_s satisfies:*

$$d_s > \frac{2\ln N - \ln\delta}{C(\epsilon)}, \quad \textit{where } C(\epsilon) = \frac{\epsilon^2}{4} - \frac{\epsilon^3}{6},$$

then for a standard Gaussian random matrix $\mathbf{G} \in \mathbb{R}^{d_v \times d_s}$, with probability at least $1-\delta$, all pairwise distances are preserved:

$$\begin{aligned} (1-\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2 &\leq \frac{1}{d_s}\|(\mathbf{x}_i - \mathbf{x}_j)\mathbf{G}\|^2 \\ &\leq (1+\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2 \end{aligned}$$

for all distinct pairs $i, j \in \{1, \ldots, N\}$. Equivalently, for the normalized projector $\mathbf{P} = \mathbf{G}/\sqrt{d_s}$, the same event is

$$(1-\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2 \leq \|(\mathbf{x}_i - \mathbf{x}_j)\mathbf{P}\|^2 \leq (1+\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2,$$

since $d_s^{-1}\|(\mathbf{x}_i - \mathbf{x}_j)\mathbf{G}\|^2 = \|(\mathbf{x}_i - \mathbf{x}_j)\mathbf{P}\|^2$.

Proof. By the bounds given in Proposition. B.2, we know that for arbitrary pair of points x_i, x_j in the set A and the unit vector $\mathbf{x} = \frac{x_i - x_j}{\|x_i - x_j\|}$,

$$\begin{aligned} \mathbb{P}\Big(\|\mathbf{x}\mathbf{G}\|^2 \geq d_s(1-\epsilon)\Big) &= \mathbb{P}\Big(\frac{1}{d_s}\|(\mathbf{x}_i - \mathbf{x}_j)\mathbf{G}\|^2 \geq (1-\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2\Big) \\ &\geq 1 - e^{-\frac{d_s}{4}\epsilon^2}. \end{aligned}$$

Similarly, we have

$$\begin{aligned}\mathbb{P}\Big(\|\mathbf{x}\mathbf{G}\|^2 \leq d_s(1+\epsilon)\Big) &= \mathbb{P}\bigg(\frac{1}{d_s}\|(\mathbf{x}_i-\mathbf{x}_j)\mathbf{G}\|^2 \\ &\qquad \leq (1+\epsilon)\|\mathbf{x}_i-\mathbf{x}_j\|^2\bigg) \\ &\geq 1-e^{-\frac{d_s}{2}\left(\epsilon^2/2-\epsilon^3/3\right)}.\end{aligned}$$

Therefore, by Boole's inequality, we have

$$\begin{aligned}\mathbb{P}\Big((1-\epsilon)\|\mathbf{x}_i-\mathbf{x}_j\|^2 \leq \frac{1}{d_s}\|(\mathbf{x}_i-\mathbf{x}_j)\mathbf{G}\|^2 \leq (1+\epsilon)\|\mathbf{x}_i-\mathbf{x}_j\|^2\Big) \\ \geq 1-e^{-\frac{d_s}{2}(\epsilon^2/2-\epsilon^3/3)}-e^{-\frac{d_s}{4}\epsilon^2} \\ \geq 1-2e^{-\frac{d_s}{2}(\epsilon^2/2-\epsilon^3/3)}.\end{aligned} \tag{13}$$

Let $C(\epsilon)=\frac{(\epsilon^2/2-\epsilon^3/3)}{2}$. The complementary event of inequality (13) is bounded by

$$\mathbb{P}(\text{Single Pair Fails}) \leq 2e^{-d_s C(\epsilon)}.$$

To guarantee distance preservation for the set $A=\{x_1,\ldots,x_n\}$ in $\mathbb{R}^C$, we consider all pairwise distances. The total number of pairs is $\binom{N}{2} < \frac{N^2}{2}$. By applying the Boole's inequality, the probability that any of these pairs suffers from a large distortion is bounded by the sum of their individual probabilities:

$$\begin{aligned}\mathbb{P}(\text{Any Pair Fails}) &\leq \sum_{\text{all pairs}} \mathbb{P}(\text{Single Pair Fails}) \\ &\leq N^2 \cdot e^{-d_s C(\epsilon)}.\end{aligned} \tag{14}$$

To ensure the projection fails with low probability δ, we require $N^2 \cdot e^{-d_s C(\epsilon)} < \delta$. Taking the logarithm of both sides and solving for d_s yields:

$$d_s > \frac{(2\ln N - \ln \delta)}{C(\epsilon)}. \tag{15}$$

□

We can see the standard methods to derive Johnson-Lindenstrauss inequalities employ the non-asymptotic analytical results regarding tail deviations. However, optimizing the Johnson-Lindenstrauss inequalities need not be limited to this regime; significant improvements can also be derived by incorporating exact asymptotic properties (specifically, the polynomial pre-factors). To the best of our knowledge, our work is the first to exploit such asymptotic refinements to tighten the JL bound, a direction previously unexplored in the literature.

Drawing inspiration from the *Bahadur-Rao Theorem* (Bahadur & Rao, 1960), we propose this theoretical refinement. Rather than providing a strict upper bound immediately, this fundamental result in large deviation theory establishes an exact asymptotic expansion of the tail probability. For the sum $S_k=\sum_{i=1}^k X_i$ of i.i.d. random variables X_i with $\mathbb{E}(X_i)=\mu$, the theorem establishes the following exact asymptotic expansion:

$$\mathbb{P}\left(\frac{S_k}{k} \geq \mu+\epsilon\right) \sim \frac{1}{\sqrt{2\pi k}\cdot\sigma_\lambda\cdot\lambda^*}\cdot\exp(-k\cdot I(\epsilon)), \tag{16}$$

where ϵ is a deviation. Here, the notation $\sim$ denotes asymptotic equivalence, implying that the ratio of the two sides converges to 1 as the dimension $k\to\infty$. The components are defined as follows:

- $I(\epsilon)$ is the rate function, corresponding to the Kullback-Leibler divergence that governs the exponential decay.
- λ^* represents the optimal parameter for the Cramér transform (minimizing the moment generating function).
- σ_λ denotes the standard deviation of the associated twisted distribution under the parameter λ^*.

While the classical Cramér's theorem provided only the exponential decay rate on a logarithmic scale, the Bahadur-Rao theorem marks a significant advance by identifying the missing algebraic pre-factor $\mathcal{O}(\frac{1}{\sqrt{k}})$. This contribution elevates the result from a loose exponential estimation to a sharp asymptotic expansion.

However, it is important to note that the Bahadur-Rao Theorem is not a strict bound; rather, it provides an asymptotic estimate for the tail probability. An interesting observation is that the bounds established in Proposition B.2 are dominated primarily by the exponential term involving d_s. Consequently, incorporating an algebraic pre-factor as the form of equation (16) represents a critical direction for refinement. We draw dual inspiration from the Bahadur-Rao theorem: we not only incorporate its algebraic pre-factor form but also leverage its proof methodology based on saddle point approximation to derive explicit non-asymptotic bounds. Specifically, we employ sharp Stirling bounds (Robbins, 1955) (which rigorously capture the saddle point behavior of the Gamma function) combined with precise estimates of the tail integrals. This approach yields the following two strict inequalities.

Theorem B.4. *Let $\mathbf{r} = \mathbf{x}\mathbf{G}$ for a fixed unit vector $\mathbf{x}$, so that $\|\mathbf{r}\|^2 \sim \chi^2_{d_s}$. Let $I(x)$ denote the rate function of a Chi-squared distribution with 1 degree of freedom. The following two inequalities hold:*

1) For any $\epsilon \in (0, +\infty)$, the upper tail inequality satisfies:

$$\mathbb{P}(\|\mathbf{r}\|^2 \geq d_s(1+\epsilon)) \leq \frac{1}{\sqrt{\pi}} \cdot \frac{1}{(\epsilon + 2/d_s)\sqrt{d_s}} \cdot e^{-d_s I(1+\epsilon)} \cdot e^{-\frac{1}{6d_s+1}}. \tag{17}$$

2) For any $\epsilon \in (2/d_s, 1)$, the lower tail inequality satisfies:

$$\mathbb{P}(\|\mathbf{r}\|^2 \leq d_s(1-\epsilon)) \leq \frac{1}{\sqrt{\pi}} \cdot \frac{\sqrt{d_s}}{\sqrt{(d_s\epsilon - 2)^2 + 4}} \cdot e^{-d_s I(1-\epsilon)} \cdot e^{-\frac{1}{6d_s+1}}. \tag{18}$$

Proof. The probability density function of a Chi-squared distribution with d_s degrees of freedom is given by:

$$f(t) = \frac{1}{2^{d_s/2}\Gamma(d_s/2)} t^{d_s/2-1} e^{-t/2}.$$

Proof of (17) (Upper Tail): The upper tail probability is formulated as

$$\mathbb{P}(\|\mathbf{r}\|^2 \geq x) = C_{\text{norm}}(d_s) \int_x^\infty t^{d_s/2-1} e^{-t/2}\, dt,$$

where $C_{\text{norm}}(d_s) = \frac{1}{2^{d_s/2}\Gamma(d_s/2)}$ and $x = d_s(1+\epsilon)$. First, to bound the Gamma function within $C_{\text{norm}}(d_s)$, we employ the sharp Stirling formula derived by (Robbins, 1955)

$$\Gamma(z) > \sqrt{2\pi} z^{z-1/2} e^{-z} e^{\frac{1}{12z+1}}.$$

Substituting $z = d_s/2$ into $C_{\text{norm}}(d_s)$ yields:

$$\begin{aligned} C_{\text{norm}}(d_s) &< \frac{1}{2^{d_s/2} \cdot \sqrt{2\pi}(d_s/2)^{d_s/2-1/2} e^{-d_s/2}} \cdot e^{-\frac{1}{6d_s+1}} \\ &= \frac{\sqrt{d_s}}{2\sqrt{\pi}} \cdot \left(\frac{e}{d_s}\right)^{d_s/2} \cdot e^{-\frac{1}{6d_s+1}}. \end{aligned}$$

Next, we bound the integral

$$J(x) = \int_x^\infty t^{\nu-1} e^{-t/2}\, dt$$

with $\nu = d_s/2$. Let $u = t^{\nu-1}$ and $v = -2e^{-t/2}$, which implies $du = (\nu-1)t^{\nu-2}dt$ and $dv = e^{-t/2}dt$. Applying integration by parts, we obtain:

$$J(x) = 2x^{\nu-1} e^{-x/2} + 2(\nu-1) \int_x^\infty t^{\nu-2} e^{-t/2} dt.$$

To estimate the integral on the right-hand side, we observe that $t \geq x$, which implies $1/t \leq 1/x$. Applying this bound allows us to factor $1/x$ out of the integral:

$$\begin{aligned}\int_x^\infty t^{\nu-2}e^{-t/2}\,dt &= \int_x^\infty \frac{1}{t}\cdot t^{\nu-1}e^{-t/2}\,dt \\ &\leq \frac{1}{x}\int_x^\infty t^{\nu-1}e^{-t/2}\,dt \\ &= \frac{1}{x}J(x).\end{aligned}$$

This leads to the recursive inequality:

$$J(x) \leq 2x^{\nu-1}e^{-x/2} + (2\nu-2)\cdot\frac{1}{x}J(x).$$

Since $x > 2\nu - 2$, solving for $J(x)$ gives:

$$\begin{aligned}J(x) &\leq \frac{2x^\nu}{x-2\nu+2}e^{-x/2} \\ &= \frac{x}{x-d_s+2}\cdot 2x^{d_s/2-1}e^{-x/2}.\end{aligned}$$

Substituting $x = d_s(1+\epsilon)$, the terms simplify to:

$$\begin{aligned}B &= \frac{x}{x-d_s+2} = \frac{d_s(1+\epsilon)}{d_s(1+\epsilon)-d_s+2} = \frac{d_s(1+\epsilon)}{d_s\epsilon+2}, \\ d_v &= 2x^{d_s/2-1}e^{-x/2} = 2\big(d_s(1+\epsilon)\big)^{d_s/2-1}e^{-d_s(1+\epsilon)/2}.\end{aligned}$$

Therefore, the upper tail probability $\mathbb{P}(\|\mathbf{r}\|^2 \geq d_s(1+\epsilon))$ is bounded by the product of the bounds on $C_{\text{norm}}(d_s)$ and $J(x)$

$$\begin{aligned}&\frac{\sqrt{d_s}}{2\sqrt{\pi}}\left(\frac{e}{d_s}\right)^{d_s/2}\cdot B\cdot d_v\cdot e^{-\frac{1}{6d_s+1}} \\ &\qquad = \frac{1}{\sqrt{\pi d_s}}\cdot\mathbf{M}\cdot e^{-\frac{1}{6d_s+1}}.\end{aligned}$$

where $\mathbf{M} = ((1+\epsilon)^{d_s/2-1}\cdot e^{-d_s\epsilon/2}\cdot\frac{d_s(1+\epsilon)}{d_s\epsilon+2})$. As we know, the rate function of the Chi-squared distribution is $I(x) = \frac{1}{2}(x-1-\ln x)$, thereby

$$\exp(-d_sI(1+\epsilon)) = (1+\epsilon)^{d_s/2}e^{-d_s\epsilon/2}.$$

Then we can obtain a bound for $\mathbf{M}$ as

$$\mathbf{M} = e^{-d_sI(1+\epsilon)}\cdot\frac{1}{\epsilon+2/d_s}.$$

Finally we obtain the inequality (17).

Proof of (18) (Lower Tail): The lower tail probability is given by

$$\mathbb{P}(\|\mathbf{r}\|^2 \leq y) = C_{\text{norm}}(d_s)\int_0^y t^{d_s/2-1}e^{-t/2}\,dt,$$

The bound for $C_{\text{norm}}(d_s)$ has been established above. We now focus on the integral $L(y) = \int_0^y t^{d_s/2-1}e^{-t/2}\,dt$. Using the substitution $t = y(1-u)$, we have

$$L(y) = y^\nu e^{-y/2}\int_0^1 (1-u)^{\nu-1}e^{yu/2}du. \tag{19}$$

Notice that

$$\begin{aligned}e^{-d_sI(1-\epsilon)} &= \left(\frac{e}{d_s}\right)^{d_s/2}\cdot[d_s(1-\epsilon)]^{d_s/2}e^{-\frac{d_s(1-\epsilon)}{2}} \\ &= \left(\frac{e}{d_s}\right)^{d_s/2}\cdot y^\nu e^{-y/2}.\end{aligned}$$

It remains to bound the integral kernel

$$K := \int_0^1 (1-u)^{\nu-1}e^{\nu(1-\epsilon)u}\,du, \qquad \nu := \frac{d_s}{2}.$$

For every $u \in [0,1)$, the inequality $-\ln(1-u) \geq u$ implies

$$e^{\nu(1-\epsilon)u} \leq (1-u)^{-\nu(1-\epsilon)},$$

where we used $\nu(1-\epsilon) > 0$. Therefore,

$$K \leq \int_0^1 (1-u)^{\nu-1-\nu(1-\epsilon)}\, du = \int_0^1 (1-u)^{\nu\epsilon-1}\, du = \frac{1}{\nu\epsilon}.$$

Let $\lambda := \nu\epsilon - 1$. Since $\epsilon > 2/d_s$, we have $\lambda > 0$. Thus,

$$(\lambda+1)^2 - (\lambda^2+1) = 2\lambda > 0,$$

and consequently

$$K \leq \frac{1}{\lambda+1} \leq \frac{1}{\sqrt{\lambda^2+1}} = \frac{2}{\sqrt{(d_s\epsilon-2)^2+4}}.$$

Combining this estimate with the previously established bound on $C_{\mathrm{norm}}(d_s)$ and the identity

$$\left(\frac{e}{d_s}\right)^{d_s/2} y^{d_s/2} e^{-y/2} = e^{-d_s I(1-\epsilon)}, \qquad y = d_s(1-\epsilon),$$

we obtain

$$\begin{aligned}\mathbb{P}\big(\|\mathbf{r}\|^2 \leq d_s(1-\epsilon)\big) &\leq \frac{\sqrt{d_s}}{2\sqrt{\pi}}\, e^{-d_s I(1-\epsilon) - \frac{1}{6d_s+1}}\, K \\ &\leq \frac{\sqrt{d_s}}{\sqrt{\pi}\sqrt{(d_s\epsilon-2)^2+4}}\, e^{-d_s I(1-\epsilon) - \frac{1}{6d_s+1}}.\end{aligned}$$

This proves the lower-tail bound in Eq. equation 18. Then we finally obtain the bound in the inequality (18). □

Applying the tail bounds to the normalized projector. For any fixed nonzero vector $\mathbf{v} \in \mathbb{R}^{d_v}$ chosen independently of $\mathbf{G}$,

$$\frac{d_s\|\mathbf{v}\mathbf{P}\|^2}{\|\mathbf{v}\|^2} = \frac{\|\mathbf{v}\mathbf{G}\|^2}{\|\mathbf{v}\|^2} \sim \chi^2_{d_s}.$$

Consequently, denoting the right-hand sides of Eqs. equation 17 and equation 18 by $\mathcal{B}_{\mathrm{upper}}(d_s,\epsilon)$ and $\mathcal{B}_{\mathrm{lower}}(d_s,\epsilon)$, respectively, we obtain

$$\begin{aligned}\mathbb{P}\big(\|\mathbf{v}\mathbf{P}\|^2 \geq (1+\epsilon)\|\mathbf{v}\|^2\big) &\leq \mathcal{B}_{\mathrm{upper}}(d_s,\epsilon), \\ \mathbb{P}\big(\|\mathbf{v}\mathbf{P}\|^2 \leq (1-\epsilon)\|\mathbf{v}\|^2\big) &\leq \mathcal{B}_{\mathrm{lower}}(d_s,\epsilon),\end{aligned}$$

within the parameter ranges of the corresponding bounds. These statements concern squared-norm distortion under $\mathbf{P}$; the auxiliary vector $\mathbf{r}$ is used only for the unscaled Gaussian tail calculations.

Now we compare the bounds in Proposition B.2 and Theorem B.4. The former one relies on a truncated third-order Taylor expansion ($I_{\text{Taylor}} \approx \epsilon^2/4 - \epsilon^3/6$), which underestimates the decay rate for non-negligible distortions. In contrast, Theorem B.4 utilizes the exact rate function of the Chi-squared distribution (e.g. $I(1+\epsilon) = (\epsilon - \ln(1+\epsilon))/2$). Since the exact rate retains all higher-order positive terms

$$\frac{1}{2}(\epsilon - (\epsilon - \frac{\epsilon^2}{2} + \frac{\epsilon^3}{3} - \frac{\epsilon^4}{4}\dots)) = \frac{\epsilon^2}{4} - \frac{\epsilon^3}{6} + \frac{\epsilon^4}{8} - \dots$$

that are discarded in the Taylor approximation, it yields a consistently faster exponential decay ($I > I_{\text{Taylor}}$). On the other hand, bounds in Proposition B.2 implicitly assume a constant pre-factor of $O(1)$. By leveraging the sharp Stirling formula , Theorem B.4 introduces algebraic pre-factors of order $O(1/\sqrt{d_s})$. For typical high-dimensional settings (e.g., $d_s \approx 300$), this term reduces the probability bound by a factor of approximately $\sqrt{\pi d_s} \approx 30$, providing a significant improvement. Consequently, we can use the bounds in theorem B.4 to improve the inequality (13), which directly leads to a better version of Theorem B.3. We omit the process.

B.2 GRID-CELL SEMANTIC PURITY: PRESERVATION OF CLUSTER SEPARATION

Throughout this subsection and the remaining grid analyses, $\mathbf{P} = \mathbf{G}/\sqrt{d_s}$ denotes the normalized projector defined in Appendix B.1. The guidance keys are $y_i = x_i\mathbf{P}$, and channel-wise min–max normalization subsequently produces the coordinates z_i. These two normalization operations are distinct. The range vector $\mathbf{W}(\mathbf{P})$ is computed from the guidance keys before min–max normalization. Consequently, the realized grid ambiguity diameter $s_r\|\mathbf{W}(\mathbf{P})\|$ depends on the random projection. In Theorem B.6, τ_{grid} instead denotes a deterministic upper bound selected before drawing $\mathbf{P}$, with an explicitly controlled exceedance probability.

While the sharpened bounds in Appendix B.1 significantly tighten the theoretical constraints for strict isometry, relying on global distance preservation (Theorem B.3) remains overly conservative for our streaming setting. The argument in Theorem B.3 enforces a union bound over all $\binom{N}{2}$ pairwise distances, implying a projection dimension d_s that scales logarithmically with the total token count N. In long-form video understanding where N grows indefinitely, this rigid constraint prevents the use of extremely low dimensions. To overcome this bottleneck, we shift our theoretical objective from strict geometric isometry to semantic separability. We posit that for token merging, it is unnecessary to preserve the exact distance between every pair of points; instead, it suffices to ensure that distinct semantic clusters do not collide within the grid. This relaxation allows us to bypass the heavy penalty of the global union bound and derive a theory tailored for Gram-anchored grids.

In this section, we apply our theoretical improvement of Johnson-Lindenstrauss Inequality to build our bilateral grid random projection theory. The motivation to introduce the grid setting is to improve the bound of d_s in the Theorem B.3 for find a practical d_s small enough. Even equipped with the refined bounds from Theorem B.4, the proof of the Theorem B.3 adhere to the standard Johnson-Lindenstrauss proof, which requires preserving all pairwise Euclidean distances, thereby imposing a heavy penalty on the projection dimension d_s via the union bound $\ln(N^2)$. To achieve further compression, we shift our focus to refining the problem definition tailored for BGStream.

Before analyzing the Gram-anchored Grid setting, we must have some necessary assumptions for the structure of the input data, which comprise a subset of the high-dimensional feature space. Our theoretical framework relies on the *Manifold Hypothesis*, which posits that high-dimensional real-world data (as visual features $\tilde{\mathbf{V}}_t$ defined in Sec. 3.1) are not uniformly distributed but concentrate near low-dimensional manifolds corresponding to semantic concepts. Specifically, we model the input dataset A, which consists of all flattened visual feature vectors form $\tilde{\mathbf{F}}$ (see Sec. 4.1) across all spatial locations across all temporal frames, as a union of K disjoint semantic clusters (or manifolds), such as “sky," “car," or “grass". We assume these clusters possess inherent geometric separability: points within the same semantic category share high similarity (small intra-class variance, effectively bounded by a compact radius R), while strictly distinct categories are separated by a non-zero margin in the original high-dimensional space.

Under these hypotheses, we consider BGStream’s operation of compressing these high-dimensional pixel-shuffled features into a lower-dimensional guidance space, and subsequently using the projected guidance keys to splat (or assign) visual tokens into a quantized grid. This pipeline eliminates spatiotemporal redundancy by merging similar tokens. However, a critical question arises: how does this dimensionality reduction affect the fidelity of the splatting process compared to the original high-dimensional space?

To answer this, we must recognize that our previous theoretical efforts (Theorem B.3), which strictly enforced pairwise distance preservation (isometry) for every pair of points, are ill-suited to the discretization process and overly restrictive for our goal. First, strict isometry is ill-suited because the mapping of continuous manifolds onto a discrete grid introduces inevitable quantization artifacts—specifically aliasing and boundary discontinuities. These artifacts render continuous distance guarantees functionally misaligned with the discrete nature of the output. Second, enforcing global isometry constitutes an overly restrictive constraint, as preserving the exact distance between every pair of points is unnecessary in the context of grid-based splatting. For our purpose, it suffices to ensure that the projection does not catastrophically disrupt the geometric structure of the clusters.

Accordingly, we relax the rigid geometric constraints and adopt a more pragmatic objective tailored to our data hypothesis. Ideally, a perfect projection would serve to splat every constituent point of a semantic cluster into a single, unique grid cell, thereby achieving optimal compression. While such a single-cell concentration is implausible, we must instead establish a rigorous criterion for

"projection success" that accommodates the inevitable dispersion of data. Specifically, the mapping of continuous manifolds onto a discrete grid manifests in distinct geometric phenomena. We examine these consequences through dual perspectives:

- **From a Cluster-centric perspective:** We define Intra-class Splitting as the fragmentation of a single semantic cluster into multiple grid cells (i.e., the cluster fails to stay together).
- **From a Grid-centric perspective:** We define Inter-class Mixing as the collision of data points from disjoint semantic categories into the same cell (i.e., the cell fails to stay pure).

We observe that the discrete nature of the grid creates an asymmetry between the two types of geometric distortions. We elaborate on this distinction below.

Consider a compact semantic cluster (e.g., points representing "car" features in an image containing a car). If the random projection causes this cluster to cross a grid boundary, it splits into two adjacent cells. Since BGStream performs the average pooling within each cell, this merely results in two feature tokens with similar semantic values. For downstream modules (e.g., Transformers in LLMs), this redundant representation of a semantic concept is generally benign or even beneficial. This phenomenon parallels the principle of *over-segmentation* in classical computer vision (e.g., (Achanta et al., 2012)), where partitioning an object into multiple homogeneous segments is strictly preferred over *under-segmentation* (which risks mixing distinct boundaries). In our grid-based context, these "over-segmented" yet semantically pure tokens can be effectively aggregated by downstream architectures via self-attention mechanisms, which are explicitly designed to recover global context from distributed local features.

Conversely, the phenomenon of Inter-class Mixing represents a disastrous failure. This occurs when the projection dimension is insufficient to preserve the necessary separation margin, causing points from semantically disjoint clusters (e.g., "sky" and "grass" in the picture with the car mentioned above) to collapse into the same grid cell. In this scenario, the subsequent average pooling operation forces a linear combination of mutually exclusive features. Instead of a redundant representation, this produces a feature vector that lies in the semantic void between clusters (e.g., a representation that is neither blue nor green). Such tokens destroy the decision boundaries required by downstream Transformers, effectively introducing destructive noise into the context window.

Therefore,we shift the definition of task success from strict geometric isometry to *Semantic Purity Preservation*, which is defined as following. A projection map preserves semantic purity if it strictly prevents Inter-class Mixing. That is, every non-empty grid cell in the Gram-anchored Grid must contain guidance keys belonging exclusively to a single semantic cluster. Our goal is to ensure that every non-empty grid cell remains pure, containing points from a single cluster. To achieve this, the projection does not need to strictly preserve the distance between every pair of points (which requires limiting the pairwise distortion for $\binom{N}{2}$ pairs). Instead, we only need to ensure that each point does not drift beyond the safety margin of its cluster center, thereby preventing it from intruding into the domain of neighboring clusters. To rigorously quantify this safety margin, we must first establish the geometric relationship between the discrete grid indices and continuous Euclidean distances. Consistent with the BGStream implementation described in Sec. 4.1, we assume the projected keys are mapped to the unit hypercube $[0,1]^{d_s}$ via channel-wise min-max normalization before quantization. The following lemma introduces the concept of *Grid Ambiguity Diameter*, which bounds the spatial uncertainty of grid quantization and serves as the fundamental constraint for our purity analysis.

Lemma B.5. *Let $A = \{x_1, \dots, x_N\} \subset \mathbb{R}^{d_v}$ be a set of N points. Fix a realization of the normalized projector $\mathbf{P} = \mathbf{G}/\sqrt{d_s} \in \mathbb{R}^{d_v \times d_s}$, yielding guidance keys $y_i = x_i\mathbf{P} \in \mathbb{R}^{d_s}$. The following geometric implications are deterministic for this fixed projector. Define the range vector $\mathbf{W} = (W_1, \dots, W_{d_s})$ where $W_\ell = \max_{n\in\{1,\dots,N\}} y_{n,\ell} - \min_{n\in\{1,\dots,N\}} y_{n,\ell}$ represents the width of the projection in the ℓ-th coordinate. Let $s_r < 1$ be the sampling rate, and let $q_i \in \mathbb{Z}^{d_s}$ denote the quantized grid index vector for point x_i, defined by coordinate-wise flooring $q_{i,\ell} = \lfloor (z_{i,\ell}/s_r) \rfloor$. For any pair of points (x_i, x_j) in A, a grid collision (i.e., $q_i = q_j$) implies the following geometric constraints:*

1. *If two points collide in the grid index, their Euclidean distance in the guidance space is bounded by the realized grid ambiguity diameter $s_r\|\mathbf{W}\|$:*

$$q_i = q_j \implies \|y_i - y_j\| \le s_r\|\mathbf{W}\|. \tag{20}$$

2. *Fix* $\epsilon \in (0,1)$. *If the projection satisfies*

$$(1-\epsilon)\|x_i - x_j\|^2 \le \|y_i - y_j\|^2,$$

then a collision implies

$$q_i = q_j \implies \|x_i - x_j\| \le \frac{s_r \|\mathbf{W}\|}{\sqrt{1-\epsilon}}. \tag{21}$$

3. *Equivalently, under the same lower-tail Johnson–Lindenstrauss condition,*

$$\|x_i - x_j\| > \frac{s_r \|\mathbf{W}\|}{\sqrt{1-\epsilon}} \implies q_i \ne q_j. \tag{22}$$

Proof. The proof proceeds in two steps: first, we establish the deterministic bound in the projected space; second, we lift this bound to the original space using the Johnson-Lindenstrauss property.

Let $y_i \in \mathbb{R}^{d_s}$ be the projected points. We proceed in two steps: normalization and quantization. For each dimension $\ell \in \{1, \dots, d_s\}$, we compute the dynamic range over the dataset:

$$\begin{aligned} m_\ell &:= \min_{n \in \{1,\dots,N\}} y_{n,\ell}, \\ M_\ell &:= \max_{n \in \{1,\dots,N\}} y_{n,\ell}, \\ W_\ell &:= M_\ell - m_\ell. \end{aligned}$$

We define the normalized coordinate $z_{i,\ell} \in [0,1]$ as

$$z_{i,\ell} := \begin{cases} \frac{y_{i,\ell} - m_\ell}{W_\ell}, & \text{if } W_\ell > 0 \\ 0, & \text{if } W_\ell = 0. \end{cases}$$

Rearranging this, we can express the projected coordinate as: $y_{i,\ell} = z_{i,\ell} \cdot W_\ell + m_\ell$. Let $s_r \in (0,1)$ be sampling rate. We define the integer grid index vector $q_i \in \mathbb{Z}^{d_s}$ using the floor function as

$$q_{i,\ell} := \lfloor z_{i,\ell}/s_r \rfloor, \quad q_i = (q_{i,1}, \dots, q_{i,d_s}).$$

For any index $q \in \{0, \dots, 1/s_r\}^{d_s}$, define the bucker as a set

$$B(q) := \{i \in \{1, \dots, N\} |\, q_i = q\}.$$

Assume two points i and j fall into the same bucket ($q_i = q_j$). We analyze each dimension ℓ separately:

Case 1:$W_\ell = 0$. Since the width is zero, all points have the same value in this dimension: $y_{i,\ell} = y_{j,\ell} = m_\ell$. Thus, $|y_{i,\ell} - y_{j,\ell}| = 0$.The bound $|y_{i,\ell} - y_{j,\ell}| < W_\ell \cdot s_r$ holds trivially.

Case 2: $W_\ell > 0$. Since $q_{i,\ell} = q_{j,\ell} = c$, we have:

$$\begin{aligned} & c \le \frac{z_{i,\ell}}{s_r} < c+1 \\ & \quad \text{and} \quad c \le \frac{z_{j,\ell}}{s_r} < c+1 \\ \implies & c \cdot s_r \le z_{i,\ell} < (c+1) \cdot s_r \\ & \quad \text{and} \quad c \cdot s_r \le z_{j,\ell} < (c+1) \cdot s_r. \end{aligned}$$

This implies the normalized distance is bounded by $|z_{i,\ell} - z_{j,\ell}| < s_r$. Converting back to projected space, we have

$$|y_{i,\ell} - y_{j,\ell}| = W_\ell \cdot |z_{i,\ell} - z_{j,\ell}| < W_\ell \cdot s_r.$$

Squaring and summing over all dimensions $\ell \in \{1, \dots, d_s\}$:

$$\begin{aligned} \|\mathbf{y}_i - \mathbf{y}_j\|^2 &= \sum_{\ell=1}^{d_s} (y_{i,\ell} - y_{j,\ell})^2 \\ &< \sum_{\ell=1}^{d_s} (W_\ell \cdot s_r)^2 \\ &= s_r^2 \sum_{\ell=1}^{d_s} W_\ell^2. \end{aligned}$$

Assume the projection $\mathbf{P}$ satisfies the lower-tail Johnson-Lindenstrauss property for the pair (i, j) with distortion parameter $\epsilon \in (0, 1)$:

$$(1-\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2 \le \|\mathbf{y}_i - \mathbf{y}_j\|^2.$$

Taking the square root we obtain the inequality (20).

Assume the projection $\mathbf{P}$ satisfies the lower-tail Johnson-Lindenstrauss property for the pair (i, j) with distortion parameter $\epsilon \in (0, 1)$ such that $(1-\epsilon)\|x_i - x_j\|^2 \le \|y_i - y_j\|^2$. Substituting the deterministic bound (20), we have the inequality (21)

$$(1-\epsilon)\|\mathbf{x}_i - \mathbf{x}_j\|^2 \le \|\mathbf{y}_i - \mathbf{y}_j\|^2 < (s_r\|\mathbf{W}\|)^2 \implies \|\mathbf{x}_i - \mathbf{x}_j\| < \frac{1}{\sqrt{1-\epsilon}} \cdot s_r\|\mathbf{W}\|.$$

Taking the contrapositive, we obtain the inequality (22).

□

Lemma B.5 relates grid collisions to the realized diameter $s_r\|\mathbf{W}(\mathbf{P})\|$. To obtain a probabilistic purity guarantee, we first control this random diameter by a deterministic bound. We then combine the grid-width, centroid-contraction, and point-expansion failure probabilities.

Theorem B.6. *Assume the following conditions.*

- ***Data Structure:*** *Let the fixed dataset $A = \{x_1, \dots, x_N\}$ be partitioned into $K \ge 2$ semantic clusters $C_1, \dots, C_K$, with centers $c_1, \dots, c_K$, where $K \ll N$. The data, cluster assignments, and centers are independent of the random projection.*
- ***Geometry:*** *Define the maximum intra-cluster radius and minimum inter-cluster separation as*

  $$R := \max_k \max_{x \in C_k} \|x - c_k\|, \qquad \Delta := \min_{j \ne k} \|c_j - c_k\|.$$

  Assume $R > 0$ and $\Delta > 0$.
- ***Projection and Grid Parameters:*** *Fix $d_s \ge 3$, $s_r \in (0, 1)$, and $\delta \in (0, 1)$ before drawing $\mathbf{P} = \mathbf{G}/\sqrt{d_s}$, where $G_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$. Let $\mathbf{W}(\mathbf{P})$ be the range vector of the projected dataset, as defined in Lemma B.5. Choose a deterministic grid-diameter bound $\tau_{\text{grid}} > 0$, independently of the realized projection, such that*

  $$\mathbb{P}_{\mathbf{P}}(s_r\|\mathbf{W}(\mathbf{P})\| > \tau_{\text{grid}}) \le \frac{\delta}{2}. \tag{23}$$

- ***Geometric Feasibility:*** *Assume*

  $$\Delta > \tau_{\text{grid}} + 2R,$$

  and define the deterministic critical tolerance

  $$\epsilon_{\text{crit}} := \frac{\Delta - \tau_{\text{grid}}}{2R} - 1.$$

  Before drawing $\mathbf{P}$, choose $\epsilon_g \in (2/d_s, 1)$ such that

  $$\epsilon_p(\epsilon_g) := \frac{(1-\epsilon_g)\Delta - \tau_{\text{grid}}}{2R} - 1 = \epsilon_{\text{crit}} - \frac{\Delta}{2R}\epsilon_g > 0.$$

Let $\mathcal{B}_{\text{lower}}$ and $\mathcal{B}_{\text{upper}}$ denote the respective right-hand sides of the lower- and upper-tail bounds in Theorem B.4. For later reference, define the deterministic safety radius

$$r_{\text{safe}} := \frac{(1-\epsilon_g)\Delta - \tau_{\text{grid}}}{2} = R\big(1 + \epsilon_p(\epsilon_g)\big).$$

Define the macro-level failure probability by

$$\mathbb{P}_{\text{macro}} := \mathbb{P}(\exists\, j < k : \|(c_j - c_k)\mathbf{P}\|_2 < (1-\epsilon_g)\|c_j - c_k\|_2)\,,$$

and the worst-case single-token drift probability by

$$\mathbb{P}_{\text{micro}} := \max_{1 \le k \le K} \max_{x \in C_k} \mathbb{P}(\|(x - c_k)\mathbf{P}\|_2 \ge r_{\text{safe}})\,.$$

Both probabilities are defined under the original, unconditioned Gaussian projection law. The tail bounds in Theorem B.4 imply

$$\mathbb{P}_{\text{macro}} \le \binom{K}{2} \mathcal{B}_{\text{lower}}(d_s, \epsilon_g), \qquad \mathbb{P}_{\text{micro}} \le \mathcal{B}_{\text{upper}}\big(d_s, \epsilon_p(\epsilon_g)\big).$$

If

$$\binom{K}{2} \mathcal{B}_{\text{lower}}(d_s, \epsilon_g) + N\mathcal{B}_{\text{upper}}\big(d_s, \epsilon_p(\epsilon_g)\big) \le \frac{\delta}{2}, \tag{24}$$

then, with probability at least $1-\delta$ over the draw of $\mathbf{P}$, every non-empty grid cell contains points from a single semantic cluster.

Proof. The proof combines a deterministic separation argument with bounds on three types of failure events.

First, consider a realization satisfying

$$s_r \|\mathbf{W}(\mathbf{P})\| \le \tau_{\text{grid}}.$$

By Lemma B.5, a collision between two projected points implies

$$\|x\mathbf{P} - y\mathbf{P}\| \le s_r \|\mathbf{W}(\mathbf{P})\| \le \tau_{\text{grid}}.$$

It therefore suffices to ensure that, for every $x \in C_j$ and $y \in C_k$ with $j \ne k$,

$$\|x\mathbf{P} - y\mathbf{P}\| > \tau_{\text{grid}}. \tag{25}$$

The reverse triangle inequality gives

$$\|x\mathbf{P} - y\mathbf{P}\| \ge \|(c_j - c_k)\mathbf{P}\| - \|(x - c_j)\mathbf{P}\| - \|(y - c_k)\mathbf{P}\|. \tag{26}$$

Define the deterministic safety radius

$$r_{\text{safe}} := \frac{(1-\epsilon_g)\Delta - \tau_{\text{grid}}}{2} = R\big(1 + \epsilon_p(\epsilon_g)\big).$$

If every centroid pair satisfies

$$\|(c_j - c_k)\mathbf{P}\| \ge (1-\epsilon_g)\|c_j - c_k\|,$$

and every point satisfies

$$\|(x - c_k)\mathbf{P}\| < r_{\text{safe}}, \qquad x \in C_k,$$

then Eq. equation 26 yields

$$\|x\mathbf{P} - y\mathbf{P}\| > (1-\epsilon_g)\Delta - 2r_{\text{safe}} = \tau_{\text{grid}}.$$

Thus, no inter-cluster collision can occur when all three requirements hold.

We now bound their failures over the original, unconditioned draw of $\mathbf{P}$. Since

$$(1-\epsilon_g)^2 \le 1 - \epsilon_g, \qquad (1+\epsilon_p)^2 \ge 1 + \epsilon_p,$$

the length-based contraction and expansion failure events are contained in the corresponding squared-distance failure events. Theorem B.4 therefore supplies conservative upper bounds at the fixed tolerances ϵ_g and $\epsilon_p = \epsilon_p(\epsilon_g)$.

- **Grid-Width Failure:** By Eq. equation 23,

$$\mathbb{P}_{\mathbf{P}}(s_r \|\mathbf{W}(\mathbf{P})\| > \tau_{\text{grid}}) \le \frac{\delta}{2}.$$

- **Macro Failure:** For each pair of distinct cluster centers,

$$\mathbb{P}_{\mathbf{P}}(\|(c_j - c_k)\mathbf{P}\| < (1-\epsilon_g)\|c_j - c_k\|) \le \mathcal{B}_{\text{lower}}(d_s, \epsilon_g).$$

- **Micro Failure:** For each $x \in C_k$, define point-expansion failure as

$$\|(x - c_k)\mathbf{P}\| \geq R(1 + \epsilon_p).$$

Since $\|x - c_k\| \leq R$,

$$\mathbb{P}_{\mathbf{P}}(\|(x - c_k)\mathbf{P}\| \geq R(1 + \epsilon_p)) \leq \mathcal{B}_{\text{upper}}(d_s, \epsilon_p).$$

If $x = c_k$, this failure probability is zero.

A mixed cell can occur only if at least one of these failures occurs. The union bound therefore gives

$$\mathbb{P}_{\text{total}} \leq \mathbb{P}_{\mathbf{P}}(s_r\|\mathbf{W}(\mathbf{P})\| > \tau_{\text{grid}}) + \binom{K}{2}\mathcal{B}_{\text{lower}}(d_s, \epsilon_g) + N\mathcal{B}_{\text{upper}}\big(d_s, \epsilon_p(\epsilon_g)\big) \leq \frac{\delta}{2} + \frac{\delta}{2} = \delta.$$

□

Remark B.7. The parameter ϵ_g balances centroid contraction and point expansion. Increasing ϵ_g relaxes the centroid constraint but decreases the allowable pointwise expansion tolerance:

$$\epsilon_p(\epsilon_g) = \epsilon_{\text{crit}} - \frac{\Delta}{2R}\epsilon_g.$$

All these tolerances are deterministic because τ_{grid} is selected before the projection is drawn.

The assumption $K \ll N$ motivates a simplified point-drift analysis, but the approximation must be assessed using the explicit tail bounds. When the centroid contribution is negligible and the point-expansion bound at $\epsilon_p(\epsilon_g)$ is close to its value at ϵ_{crit}, the two-term budget in Eq. equation 24 has the leading-order approximation

$$\frac{N}{\sqrt{\pi d_s}} \cdot \frac{1}{\epsilon_{\text{crit}} + 2/d_s} \cdot \exp\left(-d_s I(1 + \epsilon_{\text{crit}}) - \frac{1}{6d_s + 1}\right) \leq \frac{\delta}{2}. \tag{27}$$

The remaining $\delta/2$ is reserved for grid-width failure. The strict $1 - \delta$ guarantee uses Eq. equation 24; Eq. equation 27 is its simplified approximation.

The restriction $\epsilon_g > 2/d_s$ is required by the refined lower-tail bound used above. For $0 < \epsilon_g \leq 2/d_s$, the argument remains valid after replacing $\mathcal{B}_{\text{lower}}(d_s, \epsilon_g)$ in the risk budget by the conservative Chernoff bound

$$\exp\big(-d_s I(1 - \epsilon_g)\big).$$

Choosing the deterministic grid-diameter bound. One sufficient construction uses the diameter of the fixed input dataset,

$$D := \max_{i<j} \|x_i - x_j\|.$$

For each pair $i < j$ and coordinate ℓ, the random variable $((x_i - x_j)\mathbf{P})_\ell$ is Gaussian with variance at most D^2/d_s. The two-sided Gaussian tail bound and a union bound over all $\binom{N}{2}d_s$ pair-coordinate combinations give

$$\mathbb{P}_{\mathbf{P}}(s_r\|\mathbf{W}(\mathbf{P})\| > \tau_{\text{grid}}) \leq N(N-1)d_s \exp\left(-\frac{\tau_{\text{grid}}^2}{2s_r^2 D^2}\right).$$

Indeed, if the norm of the range vector exceeds τ_{grid}/s_r, at least one coordinate range exceeds $\tau_{\text{grid}}/(s_r\sqrt{d_s})$.

Consequently, the deterministic choice

$$\tau_{\text{grid}} = s_r D\sqrt{2\log\left(\frac{2N(N-1)d_s}{\delta}\right)}$$

satisfies Eq. equation 23. This construction can be conservative. The geometric feasibility and projection-risk conditions must still be checked; the construction alone does not guarantee that they hold. Any sharper deterministic bound satisfying Eq. equation 23 can be used instead.

B.3 Aggregate Semantic Integrity: Statistical Robustness of Merged Tokens

Theorem B.6 addresses the event that every non-empty grid cell is semantically pure. This simultaneous requirement becomes increasingly restrictive as the number of source tokens N grows, because even a small marginal failure probability can produce a large union bound because the union bound sums the marginal failure probabilities over all N tokens, potentially yielding a large overall bound even when each individual failure probability is small. We therefore study a different aggregate objective: after all tokens have been splatted, select one cell uniformly from a fixed set of grid indices and compare its expected signal and noise counts. We also quantify how contamination causes a merged representation to deviate from its reference class centroid. We give an upper bound on the expected weighted average of the representation errors across cells. Each cell's weight is proportional to the number of tokens it contains.

Projection model and semantic reference labels. Consider $N \geq 1$ visual tokens with features $\mathbf{x}_i \in \mathbb{R}^{d_v}$, values $\mathbf{v}_i \in \mathbb{R}^d$, and semantic labels $y_i \in \{1, \ldots, K\}$. For category k, let $\mathbf{c}_k$ be its feature-space center and let $\boldsymbol{\mu}_k$ be the corresponding value-space reference centroid. The correspondence between feature and value categories is an assumption of this analysis. All tokens share the projection $\mathbf{P}$, and $\mathbf{z}_i = \mathbf{x}_i\mathbf{P}$ is the continuous guidance coordinate before normalization and quantization.

Let $\mathcal{G}$ be a fixed finite set of $L \geq 1$ grid indices, including empty cells. For each index $B \in \mathcal{G}$, let $\mathcal{C}_B(\mathbf{P})$ denote the region in the projected guidance space whose coordinates are assigned to cell B after normalization and quantization. Thus, $\mathcal{C}_B(\mathbf{P})$ describes the continuous region covered by the cell, whereas B identifies the cell by its discrete index.

Write $U_i \in \mathcal{G}$ for the cell receiving token i. For each cell B, let $\mathcal{D}_k(\mathbf{P})$ denote the quantization domain in the grid given by category k, i.e. $\mathcal{D}_k(P) := \bigcup_{i:y_i=k} C_{U_i}(P)$. In the working model considered below, we assume that $s_r\|\mathbf{W}(\mathbf{P})\|_2 \leq \tau_{\text{grid}}$. Consequently, the grid-diameter bound gives

$$\operatorname{diam}\big(\mathcal{C}_B(\mathbf{P})\big) := \sup_{\mathbf{z},\mathbf{z}' \in \mathcal{C}_B(\mathbf{P})} \|\mathbf{z} - \mathbf{z}'\|_2 \leq \tau_{\text{grid}}, \qquad B \in \mathcal{G}.$$

for every cell B. The diameter of $\mathcal{C}_B(\mathbf{P})$ is measured using the projected guidance coordinates before min–max normalization. If the grid also includes spatial coordinates, we use only the guidance coordinates to compute this diameter. We retain the deterministic grid-diameter bound τ_{grid} and safety radius r_{safe} from Theorem B.6, both fixed before the projection $\mathbf{P}$ is drawn.

We adopt a macro-separated working model occurring in Theorem B.6 in which

$$\min_{k \neq \ell} \|\mathbf{c}_k\mathbf{P} - \mathbf{c}_\ell\mathbf{P}\|_2 \geq 2r_{\text{safe}} + \tau_{\text{grid}}, \qquad r_{\text{safe}} > 0, \tag{28}$$

holds throughout the analysis. The quantities in this condition may depend on the projection. Define the open safety region

$$\mathcal{S}_k(\mathbf{P}) = \{\mathbf{z} \in \mathcal{D}_k(\mathbf{P}) : \|\mathbf{z} - \mathbf{c}_k\mathbf{P}\|_2 < r_{\text{safe}}\}.$$

For each grid cell B, the corresponding region $\mathcal{C}_B(\mathbf{P})$ can intersect at most one safety region. To see this, suppose that it intersects both $\mathcal{S}_k$ and $\mathcal{S}_\ell$, where $k \neq \ell$. Then there exist points

$$\mathbf{a} \in \mathcal{C}_B(\mathbf{P}) \cap \mathcal{S}_k, \qquad \mathbf{b} \in \mathcal{C}_B(\mathbf{P}) \cap \mathcal{S}_\ell.$$

Since both points lie in the same cell region, the diameter bound gives

$$\|\mathbf{a} - \mathbf{b}\|_2 \leq \tau_{\text{grid}}.$$

However, by the definition of the safety regions and the macro-separation condition,

$$\begin{aligned}\|\mathbf{a} - \mathbf{b}\|_2 &\geq \|\mathbf{c}_k\mathbf{P} - \mathbf{c}_\ell\mathbf{P}\|_2 - \|\mathbf{a} - \mathbf{c}_k\mathbf{P}\|_2 - \|\mathbf{b} - \mathbf{c}_\ell\mathbf{P}\|_2 \\ &> (2r_{\text{safe}} + \tau_{\text{grid}}) - 2r_{\text{safe}} = \tau_{\text{grid}},\end{aligned}$$

which is a contradiction.

We therefore assign $k_B = k$ whenever $\mathcal{C}_B(\mathbf{P}) \cap \mathcal{S}_k(\mathbf{P}) \neq \varnothing$. If the footprint intersects no safety region, use a fixed deterministic fallback label, for example $k_B =$ 'null'. The label 'null' indicates that no semantic category is assigned. This analytical reference does not assign category 1 to an unclassified cell. Thus every cell, including an empty one, has a reference category. These latent reference labels are determined by the safety-region geometry and are used only for theoretical analysis; the splatting algorithm neither computes nor uses them. For a fixed dataset and fixed geometric parameters, all reference labels are fixed once the projection $\mathbf{P}$ is drawn.

The source-token throwing model. Let $\mathcal{B}_k = \{B \in \mathcal{G} : k_B = k\}$ denote all cells assigned to category k. A token succeeds whenever it reaches any cell in its category's set. Define the failure indicator and drift event

$$E_i = \mathbf{1}\{U_i \notin \mathcal{B}_{y_i}\} = \mathbf{1}\{y_i \neq k_{U_i}\},$$
$$D_i = \{\|\mathbf{x}_i\mathbf{P} - \mathbf{c}_{y_i}\mathbf{P}\|_2 \geq r_{\text{safe}}\}.$$

If D_i does not occur, the receiving footprint intersects $\mathcal{S}_{y_i}$ at $\mathbf{z}_i$ and must have label y_i. Consequently,

$$\{E_i = 1\} \subseteq D_i. \tag{29}$$

This inclusion permits a category to occupy multiple cells and does not depend on whether the receiving cell was previously empty.

All probabilities and expectations below use the same working projection model. Let $\mathbb{P}_{\text{micro}} \in [0, 1]$ be a deterministic, valid marginal bound in that model such that

$$\mathbb{P}(D_i) \leq \mathbb{P}_{\text{micro}} \quad \text{for every fixed source index } i. \tag{30}$$

The micro-risk analysis of Theorem B.6 supplies such an input when its tail bound is valid for the probability law and safety threshold used here.

Each E_i is a Bernoulli variable with marginal parameter $p_i = \mathbb{P}(E_i = 1)$, but the variables need not be independent or identically distributed. In particular, the common projector and dependence between video tokens are allowed. Define

$$p_{\text{fail}} := \frac{1}{N}\sum_{i=1}^{N} p_i \leq \mathbb{P}_{\text{micro}}. \tag{31}$$

Equivalently, this is the failure probability of a source token chosen uniformly from the N input indices. Equation equation 31 follows from the event inclusion and the marginal micro-risk bound.

Uniform cell selection and aggregation error. For each cell B, define the realized signal, noise, and total counts

$$S_B = \sum_{i=1}^{N} \mathbf{1}\{U_i = B,\ y_i = k_B\},$$
$$F_B = \sum_{i=1}^{N} \mathbf{1}\{U_i = B,\ y_i \neq k_B\},$$
$$M_B = S_B + F_B.$$

After splatting, draw J uniformly from the fixed set $\mathcal{G}$, independently of the projection and its realized occupancies. Empty cells remain eligible and have $S_B = F_B = M_B = 0$. Our SNR statistic is the ratio of expected counts

$$\text{SNR}_{\text{unif}} := \frac{\mathbb{E}[S_J]}{\mathbb{E}[F_J]}. \tag{32}$$

We use the ratio of expected counts to measure the aggregate balance between signal and noise. The expectation of the per-cell ratio would measure a different quantity and would require additional conventions for empty or noise-free cells, where $F_J = 0$.

For $M_B > 0$, let $\bar{\mathbf{V}}_B = M_B^{-1}\sum_{i:U_i=B} \mathbf{v}_i$, matching the cell average in Eq. 11. Assume $\|\boldsymbol{\mu}_k\|_2 > 0$ and $\|\mathbf{v}_i\|_2 \leq V_{\max}$, and set

$$\kappa := \max_{1\leq k\leq K} \frac{V_{\max}}{\|\boldsymbol{\mu}_k\|_2}.$$

For every cell with $S_B > 0$, assume its signal mean satisfies

$$\frac{\|\bar{\mathbf{v}}_{\text{signal},B} - \boldsymbol{\mu}_{k_B}\|_2}{\|\boldsymbol{\mu}_{k_B}\|_2} \leq \eta_{\text{intra}}, \qquad \bar{\mathbf{v}}_{\text{signal},B} = \frac{1}{S_B}\sum_{\substack{i:U_i=B \\ y_i=k_B}} \mathbf{v}_i. \tag{33}$$

Here $V_{\max}$ and η_{intra} are deterministic constant which bound the norm of every input value vector, including both signal and noise tokens, and the relative deviation of each cell's signal mean from its reference class centroid, whenever the cell contains signal tokens. Both bounds are fixed before the

projection is drawn and apply uniformly across the cells and projection realizations considered in the analysis. The signal-dispersion condition 33 is an additional assumption on the value vectors; it does not follow from separation in the guidance space. Tokens from the same class that enter a particular cell may represent only a specific part of that class. The bound η_{intra} must therefore also account for any difference between the mean of this selected subset and the full class centroid. Define

$$e_B = \frac{\|\bar{\mathbf{V}}_B - \boldsymbol{\mu}_{k_B}\|_2}{\|\boldsymbol{\mu}_{k_B}\|_2} \quad (M_B > 0),$$

and set $e_B = 0$ for an empty cell solely to define weighted sums. We measure aggregate representation error by

$$\mathcal{E}_{\text{agg}} := \frac{\mathbb{E}[M_J e_J]}{\mathbb{E}[M_J]} = \frac{1}{N}\mathbb{E}\left[\sum_{B\in\mathcal{G}} M_B e_B\right]. \tag{34}$$

This weights each merged representation by the number of source tokens it represents. The quantity e_B measures how far the merged token in cell B is from its reference class centroid, relative to the norm of that centroid. The aggregate error $\mathcal{E}_{\text{agg}}$ first averages these cell errors with weights M_B/N, and then takes the expectation over the random projection. Thus, a cell containing more source tokens receives a larger weight, so that each source token contributes equally to the aggregate metric. Empty cells contribute nothing because their token counts are zero.

Theorem B.8. *Under the macro-separated model, safety-region label assignment, and marginal micro-risk bound above,*

$$\mathbb{E}[S_J] = \frac{N}{L}(1 - p_{\text{fail}}), \qquad \mathbb{E}[F_J] = \frac{N}{L}p_{\text{fail}}. \tag{35}$$

For $0 < p_{\text{fail}} \le \mathbb{P}_{\text{micro}} \le 1$,

$$\text{SNR}_{\text{unif}} = \frac{1 - p_{\text{fail}}}{p_{\text{fail}}} \ge \frac{1 - \mathbb{P}_{\text{micro}}}{\mathbb{P}_{\text{micro}}}. \tag{36}$$

If the value-space assumptions also hold, every occupied cell obeys the deterministic bound

$$e_B \le \frac{S_B}{M_B}\eta_{\text{intra}} + \frac{F_B}{M_B}(\kappa + 1), \tag{37}$$

and the aggregate error satisfies

$$\mathcal{E}_{\text{agg}} \le \eta_{\text{intra}}(1 - p_{\text{fail}}) + (\kappa + 1)p_{\text{fail}} \le \eta_{\text{intra}} + (\kappa + 1)\mathbb{P}_{\text{micro}}. \tag{38}$$

Proof. Each source token contributes to exactly one cell, so for every realization,

$$\sum_{B\in\mathcal{G}} F_B = \sum_{i=1}^{N} E_i, \qquad \sum_{B\in\mathcal{G}} S_B = N - \sum_{i=1}^{N} E_i, \qquad \sum_{B\in\mathcal{G}} M_B = N.$$

Linearity of expectation gives expected total noise and signal counts Np_{fail} and $N(1 - p_{\text{fail}})$, respectively. Uniform, independent selection of the index J yields

$$\mathbb{E}[F_J] = \frac{1}{L}\mathbb{E}\left[\sum_{B\in\mathcal{G}} F_B\right] = \frac{N}{L}p_{\text{fail}},$$

$$\mathbb{E}[S_J] = \frac{1}{L}\mathbb{E}\left[\sum_{B\in\mathcal{G}} S_B\right] = \frac{N}{L}(1 - p_{\text{fail}}).$$

Taking the ratio and using the monotonicity of $(1 - p)/p$ proves Eq. equation 36. These steps use marginal expectations, not a binomial distribution for any receiving cell.

For an occupied cell with $S_B > 0$, decompose

$$\bar{\mathbf{V}}_B - \boldsymbol{\mu}_{k_B} = \frac{S_B}{M_B}(\bar{\mathbf{v}}_{\text{signal},B} - \boldsymbol{\mu}_{k_B}) + \frac{1}{M_B}\sum_{\substack{i:U_i=B \\ y_i\neq k_B}} (\mathbf{v}_i - \boldsymbol{\mu}_{k_B}).$$

The signal term is bounded using Eq. equation 33. Each noise token satisfies

$$\frac{\|\mathbf{v}_i - \boldsymbol{\mu}_{k_B}\|_2}{\|\boldsymbol{\mu}_{k_B}\|_2} \le \frac{V_{\max}}{\|\boldsymbol{\mu}_{k_B}\|_2} + 1 \le \kappa + 1.$$

The triangle inequality therefore proves Eq. equation 37. If $S_B = 0$, all received tokens are noise and the same inequality follows directly from their norm bounds, without defining a signal mean. Multiplying the local inequality by M_B, summing over cells, and taking expectations gives

$$\begin{aligned}\frac{1}{N}\mathbb{E}\left[\sum_B M_B e_B\right] &\le \frac{\eta_{\text{intra}}}{N}\mathbb{E}\left[\sum_B S_B\right] + \frac{\kappa+1}{N}\mathbb{E}\left[\sum_B F_B\right] \\ &= \eta_{\text{intra}}(1 - p_{\text{fail}}) + (\kappa+1)p_{\text{fail}}.\end{aligned}$$

The last inequality in Eq. equation 38 uses $p_{\text{fail}} \le \mathbb{P}_{\text{micro}}$. □

Remark B.9 (Interpretation and numerical bound.). The expected signal and noise counts both scale as N/L, so their ratio has no union-bound factor N.

To evaluate the SNR bound numerically, let $b(d_s, \theta)$ be a valid deterministic upper bound on $\mathbb{P}_{\text{micro}}$ in the working model. In the squared-norm tail convention, one may write

$$\begin{aligned} b(d_s,\theta) &= \min\{1, \mathcal{B}_{\text{upper}}(d_s,\theta)\}, \\ \text{SNR}_{\text{unif}} &\ge \frac{1 - b(d_s,\theta)}{b(d_s,\theta)}. \end{aligned} \tag{39}$$

The substitution requires the preceding tail analysis to establish that b bounds the same drift event and probability model. If $r_{\text{safe}} = R(1+\epsilon_p)$, the squared-norm tolerance is

$$\theta = (r_{\text{safe}}/R)^2 - 1 = 2\epsilon_p + \epsilon_p^2, \tag{40}$$

rather than ϵ_p itself. If the safety threshold or normalized grid range is random, it must first be controlled to obtain the deterministic tail bound used in Eq. equation 39.

Figure 4 evaluates Eq. equation 39 for $d_s \in \{3, \dots, 12\}$ and squared-norm tolerances $\theta \in \{0.1, \dots, 0.9\}$, using the upper-tail expression

$$\mathcal{B}_{\text{upper}}(d_s,\theta) = \frac{\exp\{-d_s I(1+\theta) - 1/(6d_s+1)\}}{\sqrt{\pi d_s}\,(\theta + 2/d_s)}, \qquad I(t) = \frac{t - 1 - \log t}{2}.$$

For example, $d_s = 4$ with $\theta = 0.9$ and $d_s = 5$ with $\theta = 0.8$ yield expected-count SNR lower bounds of approximately 7.656 and 7.349, respectively, under the stated assumptions. The parameter θ measures the geometric tolerance to within-cluster expansion: larger values correspond to a larger safety radius relative to the original cluster radius. This tolerance can reflect aspects of image quality through the geometry of the extracted features. When higher-quality inputs produce more compact and better-separated semantic clusters, they allow a larger θ for a given grid scale and centroid-contraction tolerance. The curves on Figure 4 therefore illustrate how favorable semantic feature geometry can support a high aggregate expected-count SNR even at small guidance dimensions.

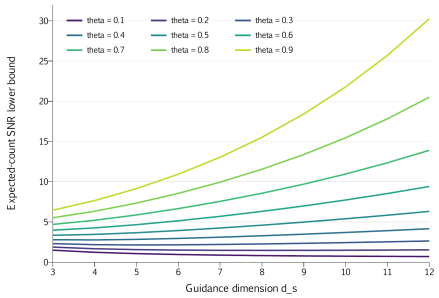
B.4 Expectation of Vacancy Rate of Bilateral Grids

To analyze the efficiency of the Gram-anchored grid in consolidating redundant information, we first define the occupancy metrics. Let V_{total} be the total capacity of the grid index space $\mathcal{U}$. After the splatting process, we define the number of occupied grid cells, denoted as V_o, as the total count of grid cells containing at least one visual token. Conversely, the number of empty grid cells, denoted as V_e, is the count of cells containing zero tokens, which naturally satisfies $V_e = V_{total} - V_o$.

We then define the vacancy rate $\mathcal{V}$ of the Gram-anchored grid as the proportion of empty cells relative to the total capacity

$$\mathcal{V} := \frac{V_e}{V_{total}} = 1 - \frac{V_o}{V_{total}}.$$

For a fixed input dataset, the vacancy rate $\mathcal{V}$ varies with the random projection through the projected coordinates, min–max normalization, and quantization. The following theorem studies occupancy under a uniform bounded-drift model, in which every token remains within a prescribed radius of its projected cluster center.



d_s	θ=0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
3	1.498	1.880	2.310	2.797	3.350	3.981	4.702	5.525	6.467
4	1.235	1.676	2.182	2.770	3.457	4.264	5.213	6.332	7.656
5	1.071	1.567	2.148	2.838	3.666	4.665	5.876	7.349	9.146
6	0.960	1.509	2.163	2.958	3.936	5.149	6.661	8.555	10.934
7	0.882	1.480	2.207	3.111	4.250	5.701	7.559	9.952	13.047
8	0.825	1.471	2.271	3.288	4.601	6.315	8.571	11.555	15.524
9	0.782	1.476	2.350	3.484	4.984	6.992	9.701	13.384	18.417
10	0.750	1.490	2.439	3.697	5.398	7.731	10.960	15.463	21.785
11	0.726	1.511	2.536	3.924	5.843	8.536	12.357	17.823	25.701
12	0.707	1.538	2.641	4.164	6.317	9.410	13.904	20.497	30.247

Figure 4: **Uniform-cell expected-count SNR.** The curves and table evaluate $(1 - b(d_s, \theta))/b(d_s, \theta)$, where b is the stated upper-tail expression capped at one. Here θ is a squared-norm drift tolerance. The statistic is $\mathbb{E}[S_J]/\mathbb{E}[F_J]$ for a cell selected uniformly from the fixed full grid, including empty cells. Its interpretation as a lower bound requires b to be a valid marginal drift bound under the macro-separated working model.

Theorem B.10. *Under the Manifold Hypothesis (Appendix B.2), assume the N samples are drawn from K disjoint semantic clusters ($K \ll N$), where each cluster has a maximum intra-class radius R in the original space. Fix a deterministic tolerance $\epsilon_p > 0$ before drawing the projection, using the same drift-radius convention as in Theorem B.6. Assume that the projection distribution considered in this theorem is supported on realizations satisfying the uniform bounded-drift condition*

$$\max_{1 \le k \le K} \max_{x_i \in C_k} \|(x_i - c_k)\mathbf{P}\|_2 \le R(1 + \epsilon_p),$$

where c_k is the center of cluster C_k. All expectations $\mathbb{E}$ in this theorem and its proof are taken under this bounded-drift projection model. Then, the expected vacancy rate $\mathbb{E}[\mathcal{V}]$ is lower-bounded by

$$\mathbb{E}[\mathcal{V}] \ge 1 - \frac{\min\left(N,\ K \cdot \mathcal{B}(s_s, s_r)\right)}{V_{total}},$$

where $\mathcal{B}(s_s, s_r)$ is the maximum number of grid cells spanned by a single semantic cluster bounded by

$$\mathcal{B}(s_s, s_r) \le \left\lfloor \frac{H}{s_s} \right\rfloor \left\lfloor \frac{W}{s_s} \right\rfloor \cdot \left\lceil \frac{2R(1 + \epsilon_p)}{s_r} \right\rceil^{d_s}.$$

Proof. By the definition of the vacancy rate $\mathcal{V} = 1 - V_o/V_{total}$, its expectation is given by the expression $\mathbb{E}[\mathcal{V}] = 1 - \mathbb{E}[V_o]/V_{total}$. To derive a lower bound for the expected vacancy rate, it is necessary to establish an upper bound for the number of occupied grid cells V_o. For a total of N input vision tokens, V_o is constrained by the sample size such that $V_o \le N$. Furthermore, under the Manifold Hypothesis, visual features are not uniformly distributed but are concentrated within K disjoint semantic clusters. By the uniform bounded-drift assumption, every projected token from cluster C_k lies within distance $R(1 + \epsilon_p)$ of its projected center $c_k\mathbf{P}$. Hence, for any $x_i, x_j \in C_k$,

$$\|(x_i - x_j)\mathbf{P}\|_2 \le \|(x_i - c_k)\mathbf{P}\|_2 + \|(x_j - c_k)\mathbf{P}\|_2 \le 2R(1 + \epsilon_p).$$

It implies that in each of the d_s semantic dimensions, the cluster can span at most $\lceil 2R(1 + \epsilon_p)/s_r \rceil$ cells. When incorporating the spatial dimensions defined in the grid index space $\mathcal{U}$, the maximum number of cells $\mathcal{B}$ that a single semantic cluster can occupy is given by

$$\mathcal{B}(s_s, s_r) \le \lfloor H/s_s \rfloor \lfloor W/s_s \rfloor \cdot \lceil 2R(1 + \epsilon_p)/s_r \rceil^{d_s}.$$

The total number of occupied cells V_o is thus bounded by the cumulative span of all K clusters, yielding $V_o \le \sum_{k=1}^{K} \mathcal{B} = K \cdot \mathcal{B}$. By combining these constraints, we obtain $V_o \le \min(N, K \cdot \mathcal{B})$. Finally, taking the expectation and substituting this upper bound into the vacancy rate formula confirms that

$$\mathbb{E}[\mathcal{V}] \ge 1 - \frac{\min\left(N,\ K \cdot \mathcal{B}(s_s, s_r)\right)}{V_{total}}.$$

□

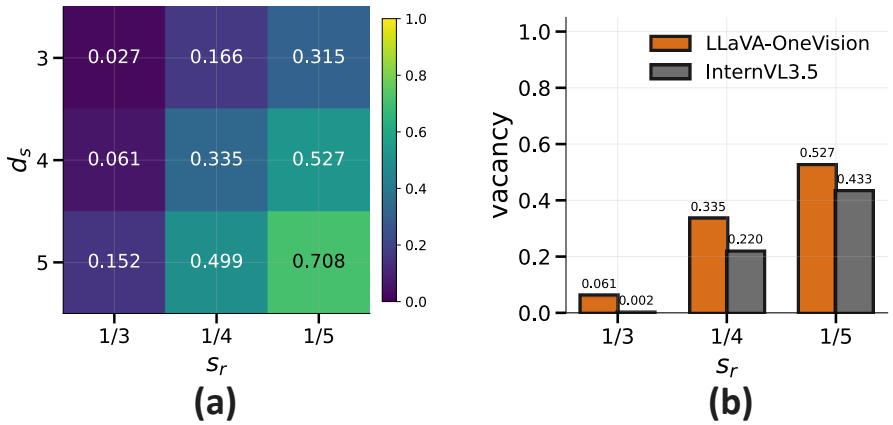


Figure 5: **Grid occupancy on StreamingBench. (a)** Vacancy rate of LLaVA-OneVision over d_s and s_r. **(b)** Vacancy rate of LLaVA-OneVision and InternVL-3.5 at d_s=4.

s_r		d_s = 3	4	5
1/3	V_o	26.3	76.0	206.2
	V_{total}	27	81	243
1/4	V_o	53.3	170.2	513.0
	V_{total}	64	256	1024
1/5	V_o	85.7	295.5	913.3
	V_{total}	125	625	3,125

Table 6: Number of non-empty grid cells (V_o) and the total cell budget ($V_{\text{total}} = \lceil 1/s_r \rceil^{d_s}$) per n_g-frame group on StreamingBench (LLaVA-OneVision; mean over 26 groups from 21 videos that form a full n_g-group).

The model-to-model discrepancy we observe is itself consistent with our analysis: it follows from how the cluster radius R and the semantic count K enter the vacancy bound. We examine this empirically below.

Grid occupancy: two regimes and backbone dependence. Our analysis bounds the number of occupied cells by $V_o \leq \min(N,\ K \cdot B)$, where B—the cells a single cluster can span—increases with the grid resolution $\lceil 1/s_r \rceil^{d_s}$ and the intra-class radius R. To probe this empirically, we measure the vacancy rate over the full sweep of $s_r \in \{1/3, 1/4, 1/5\}$ and $d_s \in \{3, 4, 5\}$ on LLaVA-OneVision, which is shown in Figure 5. The analysis reveals two regimes.

Coarse grid (low vacancy). When $\lceil 1/s_r \rceil^{d_s}$ is small (top-left, s_r=1/3, d_s=3: vacancy 0.027), the cell budget is far below the N incoming tokens, so cells saturate. The grid compresses many tokens into few cells—the strongly compressive regime BGStream operates in.

Fine grid (high vacancy). As the grid is refined (bottom-right, s_r=1/5, d_s=5: vacancy 0.708), V_o no longer fills the enlarged grid and vacancy rises monotonically along both axes. Yet this rising vacancy is *not* beneficial: each group then emits more grid tokens (V_o grows from 26.3 to 913; Table 6), loosening merging and inflating the retention ratio (Figure 3). The strong compression of a coarse grid outweighs it, so we adopt s_r=1/3.

Backbone dependence. Since B increases with the cluster radius R, features that are more diffuse (larger R) spread over more cells and leave the grid less vacant. Consistently, across all three s_r in Figure 5(b), InternVL-3.5—whose measured R is larger than LLaVA-OneVision's (215.1 vs. 180.7)—exhibits lower vacancy at d_s=4, indicating that the same mechanism governs occupancy across backbones.

C Details of Method

C.1 Attention-based Token Selection in the FIFO Buffer

In Sec. 4.3, we noted that naively retaining all $n_r \cdot n_v$ tokens in the FIFO buffer is prohibitive under the tight token budget of streaming inference. To mitigate this, we retain only the top-$\rho\%$ visual tokens per frame, ranked by an importance score derived from the visual encoder's self-attention.

Computing the importance score. A common practice is to compute token importance using the CLS token (Zhang et al., 2024b; Wang et al., 2024). On the other hand, for visual encoders without a CLS token (e.g., SigLIP (Zhai et al., 2023)), we follow prior work (Yang et al., 2025a; Shao et al., 2025) and instead compute the importance of each visual token from the *average attention it receives* from all other tokens. Specifically, let $\mathbf{A} \in \mathbb{R}^{n_v \times n_v}$ denote the head-averaged self-attention matrix at the final layer of the visual encoder. The importance score of the i-th token is

$$s_i \;=\; \frac{1}{n_v - 1} \sum_{j \neq i} \mathbf{A}_{j,i}, \tag{41}$$

which measures how strongly token i is attended to by the rest of the tokens; Using $\{s_i\}_{i=1}^{n_v}$ as the ranking criterion, we retain the top-$\rho\%$ tokens per frame within the FIFO buffer and discard the rest, yielding $\lceil \rho \cdot n_v \rceil$ tokens per buffered frame. This reduces the FIFO buffer's token consumption while preserving the most informative perceptual evidence near the query timestamp t_q. It is worth noting that this attention-based selection is applied only when feeding FIFO tokens to the LLM; the bilateral grid splatting in Sec. 4.2 operates on the full set of visual tokens.

C.2 Cross-Grid Merging under a Token Budget

In Sec. 4.4, we noted that grid tokens accumulate linearly with t_q, so the total visual context $\widehat{\mathbf{V}}_{1:t_q}$ (FIFO tokens combined with all grid tokens) may eventually exceed the LLM's context window B. To respect B throughout an arbitrarily long stream, we apply a *hierarchical cross-grid merging* step that progressively consolidates the oldest grids into coarser representations while keeping the most recent grids at finer temporal resolutions. This recency-aware schedule reflects the empirical observation that frames near t_q typically carry the most decision-relevant evidence, so older content can tolerate coarser representation.

Notation. Let $G_1, G_2, \ldots, G_T$ denote the bilateral grids accumulated up to t_q, ordered from oldest (G_1) to most recent (G_T). Each grid initially aggregates n_g consecutive frames (Sec. 4.1.1). We use the bracket notation $[m]$ to indicate that a grid covers m frames; the initial sequence thus reads $[n_g] \to [n_g] \to \cdots \to [n_g]$. For a non-empty cell at index $\mathbf{u}$, we denote by $\bar{\mathbf{v}}^i_{\mathbf{u}}$ the per-cell representative token of G_i (Eq. 10) and by $w^i_{\mathbf{u}} = \Gamma_i[\mathbf{u}]_c$ its *homogeneous-coordinate weight* (Chen et al., 2007)—i.e., the number of visual tokens splatted into that cell.

Pair-wise grid merging. The basic operation is the merging of two adjacent grids G_i and G_{i+1} into a single grid G'_i that aggregates $2m$ frames (assuming each originally covered m). For every cell index $\mathbf{u}$ shared by the two grids, the merged token is the count-weighted average

$$\begin{aligned} \bar{\mathbf{v}}^{i'}_{\mathbf{u}} &= \frac{\bar{\mathbf{v}}^i_{\mathbf{u}} \cdot w^i_{\mathbf{u}} + \bar{\mathbf{v}}^{i+1}_{\mathbf{u}} \cdot w^{i+1}_{\mathbf{u}}}{w^i_{\mathbf{u}} + w^{i+1}_{\mathbf{u}}}, \\ w^{i'}_{\mathbf{u}} &= w^i_{\mathbf{u}} + w^{i+1}_{\mathbf{u}}, \end{aligned} \tag{42}$$

following the homogeneous-coordinate convention of the bilateral grid (Chen et al., 2007). Cells appearing in only one of the two grids are carried over unchanged together with their weights. Eq. 42 extends the per-cell aggregation in Eq. 4 from *within* a single grid to *across* two adjacent grids, so the resulting G'_i remains a valid bilateral grid in homogeneous form.

Hierarchical schedule via batched pair-wise merging. Whenever the total token count $|\widehat{\mathbf{V}}_{1:t_q}|$ exceeds B, we apply Eq. 42 to *every* adjacent pair in the current grid sequence at once (non-overlapping, oldest pair first). This batched merging halves the number of grids in a single pass, freeing enough budget headroom to absorb a substantial number of incoming frames before the next budget hit. Crucially, because freshly arriving frames continue to form fine-grained $[n_g]$ grids at the

tail between budget hits, the older grids accumulate more merge passes than the recent ones across the stream's lifetime—producing a representation in which temporal granularity smoothly coarsens with distance from t_q. With n_g=8 as a running example:

- **Stage 0 (initial).** Every grid covers n_g frames: $[8] \to [8] \to \cdots \to [8]$.
- **Stage 1 (first budget hit).** All adjacent pairs are merged in one pass, halving the grid count: $[16] \to [16] \to \cdots \to [16]$.
- **Stage 2 (streaming resumes).** Newly arriving frames keep forming fresh $[n_g]$ grids at the tail until the next budget hit: $[16] \to \cdots \to [16] \to [8] \to [8] \to \cdots \to [8]$.
- **Stage 3 (next budget hit).** All adjacent pairs are again merged in one pass; the older $[16]$ grids become $[32]$ while the recent $[8]$ grids become $[16]$: $[32] \to \cdots \to [32] \to [16] \to \cdots \to [16]$.

Repeating this cycle, frames farther from t_q undergo more merge passes and end up summarized at progressively coarser temporal resolutions, while recent frames—having undergone fewer merges—retain finer resolution. No explicit "tail-protection" threshold is needed: the hierarchy emerges naturally from the interplay between batched merging and continuous tail growth, keeping $\widehat{\mathbf{V}}_{1:t_q} \leq B$ for arbitrarily long streams.

D Experiments

D.1 Additional Details of Dataset

All benchmarks we use consist of English video QA over publicly available video content, sourced primarily from YouTube and existing egocentric corpora. StreamingBench (Lin et al., 2024) and OVO-Bench (Niu et al., 2025) target online/streaming video understanding with multi-task QA covering real-time perception, sequential reasoning, and related skills. EgoSchema (Mangalam et al., 2023) is a long-form egocentric video QA benchmark built on Ego4D. LongVideoBench (Wu et al., 2024) and MLVU (Zhou et al., 2025) (we use the MLVU dev set) target long-form video understanding across diverse domains including movies, surveillance, egocentric, and instructional videos. All five are released by their original authors for research purposes and we use them without modification.

For StreamingBench that contains 2,500 questions, we evaluate on the real-time visual understanding subset, following prior work (Wang et al., 2025b; Zeng et al., 2025). For OVO-Bench containing 1,462 questions, following (Shen et al., 2026), we use the Backward Tracing and Real-Time Visual Perception categories; we exclude Forward Active Responding, as it requires deciding *when* to respond, which is orthogonal to token compression. For the offline setting, we follow (Chen et al., 2025) and sample at 0.5 FPS for videos under 30 minutes and 0.2 FPS for longer ones to avoid memory overflow. We report multiple-choice accuracy in all experiments.

D.2 Additional Details of Baselines

In this paper, we evaluate S+T compression methods within a streaming setting. Importantly, with the exception of StreamingTOM (Chen et al., 2025), the spatial-temporal compression baselines are originally designed for the *offline* setting: they partition the entire video into segments and remove temporal redundancy within each segment, presupposing access to the full video at compression time. This assumption is fundamentally at odds with streaming video understanding, where future frames are inherently unavailable. For a comprehensive comparison, we adapt these methods by treating the frames observed up to each query timestamp as the "full video" and applying their compression accordingly. Spatial compression methods, in contrast, are inherently streaming-compatible, as each frame can be processed independently upon arrival.

D.3 Additional Implementation Details

For all experiments, we use a single set of hyperparameters for BGStream: the temporal group size n_g=32, the range sampling rate s_r=1/3, and the JL projection dimension d_s=4 for grid construction; the FIFO buffer size n_r=5 and the per-frame retention ratio ρ=20% for attention-based token

Table 7: Comparison on StreamingBench Lin et al. (2024) under per-query matched retention ratios. Each baseline retains the exact same number of visual tokens as BGStream for every query.

T	Method	OP	CR	TR	ATP	EU	ACP	SU	CS	PR	CT	Overall
S	ToMe (Bolya et al., 2022) (ICLR'23)	69.65	75.00	58.57	69.23	62.89	57.95	60.57	75.39	62.96	26.06	62.64
S	VisionZip (Yang et al., 2025a) (CVPR'25)	62.87	71.88	58.88	62.18	64.78	59.09	53.25	70.35	58.33	19.15	58.84
S	FastVid (Shen et al., 2025) (NeurIPS'25)	67.21	72.66	55.76	59.29	65.41	56.53	50.41	70.98	59.26	26.06	58.80
S+T	PruneVid (Huang et al., 2025) (ACL'25)	67.48	73.44	59.81	71.47	69.81	58.81	58.94	74.45	59.26	35.64	63.52
S+T	Holitom (Shao et al., 2025) (NeurIPS'25)	66.12	**76.56**	56.39	66.67	66.04	58.52	58.54	72.24	58.33	33.51	61.64
S+T	FlashVid (Fan et al., 2026) (ICLR'26)	61.52	68.75	57.94	64.74	67.92	61.93	55.28	71.61	66.67	28.19	60.68
S+T	StreamingTom (Chen et al., 2025) (CVPR'26)	64.23	**76.56**	58.26	64.10	60.38	57.95	54.07	71.92	57.41	33.51	60.32
	BGStream ($\sim$5.5%)	**77.24**	74.22	**64.17**	**75.00**	**72.96**	**65.62**	**60.57**	**86.44**	**69.44**	26.60	**68.60**

Table 8: Comparison on OVO-Bench (Niu et al., 2025) under per-query matched retention ratios. Each baseline retains the exact same number of visual tokens as BGStream for every query (average $\sim$4.5%).

T	Method	Real-Time Visual Perception							Backward Tracing				Overall
		OCR	ACR	ATR	STU	FPD	OJR	Avg.	EPM	ASI	HLD	Avg.	
S	ToMe (Bolya et al., 2022) (ICLR'23)	44.30	51.38	**67.27**	39.33	68.32	48.37	51.02	49.16	52.03	30.65	44.37	48.15
S	VisionZip (Yang et al., 2025a) (CVPR'25)	40.27	45.87	52.73	33.71	58.42	44.02	44.28	49.83	53.38	28.49	44.37	44.32
S	FastVid (Shen et al., 2025) (NeurIPS'25)	48.32	46.79	54.55	37.64	64.36	50.54	49.10	41.75	50.00	19.89	37.24	43.98
S+T	PruneVid (Huang et al., 2025) (ACL'25)	42.28	50.46	62.73	40.45	65.35	54.35	51.14	44.78	53.38	19.89	39.46	46.10
S+T	Holitom (Shao et al., 2025) (NeurIPS'25)	48.32	44.04	60.91	36.52	65.35	47.28	48.74	45.45	53.38	24.73	41.20	45.49
S+T	FlashVid (Fan et al., 2026) (ICLR'26)	41.61	49.54	50.91	36.52	66.34	44.02	46.33	48.48	54.73	25.27	43.11	44.94
S+T	StreamingTom (Chen et al., 2025) (CVPR'26)	42.28	43.12	44.55	35.39	65.35	47.83	45.25	44.78	**56.08**	**36.02**	44.85	45.08
	BGStream ($\sim$4.5%)	**65.10**	**62.39**	63.64	**44.38**	**70.30**	**67.39**	**61.25**	**53.87**	54.73	25.81	**45.80**	**54.58**

selection (Appendix C.1); and the target token budget B=18,000, beyond which the cross-grid merging procedure (Appendix C.2) is invoked. For all baselines, we use the default configurations specified in their respective papers, and regarding the latency comparison, we additionally cap the number of visual tokens fed into the LLM at B=18,000 for every method. When combining BGStream with ReKV (Di et al., 2025), we follow ReKV's frame-retrieval mechanism that selects which frames' offloaded KV cache entries to reload based on attention scores between the query and a per-frame global feature; since BGStream aggregates n_g frames into a single grid, we replace ReKV's per-frame mean feature with the per-cell mean of visual tokens within each grid cell, demonstrating that BGStream integrates seamlessly with existing KV cache optimization frameworks. All models and benchmarks used in this work are publicly available for research purposes under their respective licenses (LLaVA series: Apache 2.0; InternVL: MIT; StreamingBench/OVO-Bench/EgoSchema/LongVideoBench/MLVU: research-use). We use the following package versions: Python 3.10, PyTorch 2.2.1, HuggingFace transformers 4.51.3, and NumPy 1.26.4.

Spatial sampling rate s_s. We set the spatial sampling rate s_s to the full patch-grid size (i.e., $\lfloor H/s_s \rfloor = \lfloor W/s_s \rfloor = 1$), so that the grid applies no spatial partitioning. Spatial information is already encoded by the visual encoder's positional embeddings; the bilateral grid therefore focuses solely on capturing the semantic structure of visual tokens through the key projection (Section 4.1). We avoid spatial partitioning, as a finer s_s would substantially increase the number of grid cells—inflating the retention ratio—without a corresponding performance benefit.

D.4 Comparison under Per-Query Matched Retention Ratios

In Sec. 5.2 of the main paper, we compared BGStream against visual token compression baselines under a fair comparison protocol in which the retention ratio of all baselines was fixed to 5%, matching the *average* retention ratio of BGStream. Since BGStream aggregates tokens into the non-empty cells of a bilateral grid—whose count varies across queries—its per-query retention ratio varies modestly around this average. To verify that BGStream retains its advantage under exactly matched per-query token counts, we conduct a controlled comparison in which each baseline is forced to retain the same number of visual tokens as BGStream for every query.

As shown in Tables 7 and 8, BGStream consistently outperforms all baselines on both StreamingBench (Lin et al., 2024) and OVO-Bench (Niu et al., 2025), with a clear margin over the strongest baseline on each dataset. These results confirm that BGStream's advantage stems from its geometry-preserving compression rather than from any per-query mismatch in retention ratios.

D.5 Detailed Performance with InternVL3.5 Backbone

We provide per-category results on StreamingBench (Table 9) and OVO-Bench (Table 10) using InternVL3.5-8B (Wang et al., 2025a) as the backbone. The trends observed with LLaVA-OneVision

Table 9: Performance comparison on StreamingBench (Lin et al., 2024) dataset using InternVL3.5 as the backbone. "T" denotes the compression type, where "S" and "S+T" indicate spatial and spatial-temporal compression, respectively. The best results are shown in bold.

T	Method	OP	CR	TR	ATP	EU	ACP	SU	CS	PR	CT	Overall
	KV Cache Optimization + Visual Token Compression											
	ReKV (Di et al., 2025) (ICLR'25)	76.69	74.22	75.39	80.13	75.47	66.76	68.70	80.76	76.85	46.28	72.80
	+ BGStream	75.61	73.44	66.67	74.68	71.70	65.91	58.94	81.39	60.19	45.74	68.80
	Visual Token Compression											
S	ToMe (Bolya et al., 2022) (ICLR'23)	68.83	80.47	61.68	66.99	66.67	61.93	56.91	65.62	57.41	49.47	63.64
S	VisionZip (Yang et al., 2025a) (CVPR'25)	65.58	80.47	57.32	63.78	62.89	63.07	53.25	64.35	50.00	49.47	61.28
S	FastVid (Shen et al., 2025) (NeurIPS'25)	65.85	79.69	56.07	67.63	69.81	65.06	54.07	65.62	58.33	**55.32**	63.36
S+T	PruneVid (Huang et al., 2025) (ACL'25)	71.54	79.69	66.04	71.79	67.92	61.93	55.69	70.03	59.26	51.60	65.92
S+T	Holitom (Shao et al., 2025) (NeurIPS'25)	68.56	78.91	60.44	72.76	67.92	60.23	57.72	71.92	61.11	50.00	65.00
S+T	FlashVid (Fan et al., 2026) (ICLR'26)	72.36	**81.25**	59.50	72.44	69.81	63.92	59.35	71.29	62.04	48.94	66.20
S+T	StreamingTom (Chen et al., 2025) (CVPR'26)	68.02	79.69	62.31	69.55	67.30	62.22	53.66	70.35	**62.96**	49.47	64.48
	BGStream	**75.07**	80.47	**67.60**	**75.32**	**71.70**	**67.90**	**66.67**	**80.44**	57.41	45.74	**70.08**

Table 10: Performance comparison on OVO-Bench (Niu et al., 2025).

T	Method	Real-Time Visual Perception							Backward Tracing				Overall
		OCR	ACR	ATR	STU	FPD	OJR	Avg.	EPM	ASI	HLD	Avg.	
	KV Cache Optimization + Visual Token Compression												
	ReKV (Di et al., 2025) (ICLR'25)	77.85	56.88	67.27	51.69	72.28	57.61	62.94	53.20	56.08	35.48	48.65	56.77
	+ BGStream	68.46	63.30	61.82	47.19	70.30	57.61	60.17	45.79	52.03	41.94	46.12	54.10
	Visual Token Compression												
S	ToMe (Bolya et al., 2022) (ICLR'23)	49.66	49.54	55.45	43.82	71.29	51.09	52.11	44.78	45.95	31.18	41.05	47.33
S	VisionZip (Yang et al., 2025a) (CVPR'25)	41.61	42.20	43.64	44.94	67.33	44.02	46.33	41.75	47.97	**36.02**	41.52	44.25
S	FastVid (Shen et al., 2025) (NeurIPS'25)	50.34	44.95	53.64	41.57	62.38	53.26	50.30	42.76	50.00	30.11	40.73	46.17
S+T	PruneVid (Huang et al., 2025) (ACL'25)	51.68	50.46	60.91	46.63	67.33	49.46	53.07	44.11	52.03	30.65	42.00	48.29
S+T	Holitom (Shao et al., 2025) (NeurIPS'25)	46.31	53.21	57.27	44.38	67.33	51.09	51.87	**45.45**	53.38	29.03	**42.47**	47.81
S+T	FlashVid (Fan et al., 2026) (ICLR'26)	48.32	52.29	**61.82**	43.26	70.30	50.54	52.71	44.78	**56.08**	20.43	40.25	47.33
S+T	StreamingTom (Chen et al., 2025) (CVPR'26)	53.02	50.46	58.18	47.19	65.35	56.52	54.39	44.44	46.62	31.18	41.05	48.63
	BGStream	**69.80**	**66.06**	60.00	**51.12**	**76.24**	**64.13**	**63.54**	**45.45**	53.38	29.03	**42.47**	**54.45**

carry over to InternVL3.5: BGStream surpasses all visual token compression baselines by substantial margins on both benchmarks, indicating that the geometry-preserving compression mechanism is robust to backbone variation in both the visual encoder and the language model. The gains are particularly pronounced on the Real-Time Visual Perception split of OVO-Bench, with substantial improvements on perception-heavy sub-tasks such as OCR, ACR, and OJR, suggesting that the compact grid representation faithfully preserves the fine-grained perceptual evidence required for present-scene recognition. BGStream also remains compatible with KV cache optimization: combining it with ReKV substantially reduces the visual token count while incurring only a modest accuracy drop relative to ReKV alone, confirming that BGStream's geometry-preserving compression integrates seamlessly with KV cache optimization regardless of backbone. Consistent with our LLaVA-OneVision observations, BGStream shows relative weakness on the Counting (CT) sub-task of StreamingBench and the Hallucination Detection (HLD) sub-task of OVO-Bench, where coarse grid aggregation may merge tokens corresponding to numerically distinct entities into a single cell.

D.6 Results with the LLaVA-Video Backbone

To further assess generality of BGStream, we evaluate it with the LLaVA-Video-7B (Zhang et al., 2024c) backbone on StreamingBench and OVO-Bench benckmarks. As shown in Table 11, we observe that BGStream achieves the best overall accuracy among all visual token compression baselines on both benchmarks, and on OVO-Bench it even surpasses ReKV, which retains all visual tokens. These results confirm that BGStream generalizes across backbones.

Table 11: Results with the LLaVA-Video (Zhang et al., 2024c) backbone.

T	Method	StreamingBench Overall	OVO-Bench Overall
	KV Cache Optimization		
	ReKV (Di et al., 2025)	73.76	55.81
	Visual Token Compression		
S	ToMe (Bolya et al., 2022)	65.64	47.67
S	VisionZip (Yang et al., 2025a)	62.48	44.46
S	FastVid (Shen et al., 2025)	62.72	47.95
S+T	PruneVid (Huang et al., 2025)	66.00	50.48
S+T	Holitom (Shao et al., 2025)	66.04	49.59
S+T	FlashVid (Fan et al., 2026)	64.92	50.21
S+T	StreamingTOM (Chen et al., 2025)	61.12	46.65
	BGStream	**67.92**	**58.14**

D.7 Distortion and Semantic Purity Metrics

In Section 5.6 of the main paper, we empirically demonstrate that BGStream effectively preserves the semantic structure among visual tokens using two metrics, **distortion** ϵ and **semantic purity**. Both metrics are computed on 32 randomly sampled StreamingBench videos. To mitigate the prohibitive cost of evaluating all pairwise distances of visual tokens, we randomly sample 50,000 token pairs per video. We describe each metric in detail below.

Distortion. The distortion metric quantifies (Dasgupta & Gupta, 2003) how much a compression method perturbs the pairwise geometry of the original visual tokens. Let $\mathbf{V}_{\text{orig}} = \{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$ denote the original visual tokens and $\mathbf{V}_{\text{comp}}$ the set of compressed tokens, with an assignment $a(\cdot)$ that maps each original token to its assigned compressed token (i.e., its centroid). For a sampled token pair (i, j), we define the original and compressed distances as

$$d_{\text{orig}}(i,j) = \|\mathbf{v}_i - \mathbf{v}_j\|_2,$$
$$d_{\text{comp}}(i,j) = \|\mathbf{v}^{\text{comp}}_{a(i)} - \mathbf{v}^{\text{comp}}_{a(j)}\|_2,$$

where $\mathbf{v}^{\text{comp}}_{a(i)}$ is the compressed token to which $\mathbf{v}_i$ is assigned. The distortion is the mean relative deviation between the two distances over all sampled pairs:

$$\text{Distortion} = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} \frac{|d_{\text{orig}}(i,j) - d_{\text{comp}}(i,j)|}{d_{\text{orig}}(i,j) + \xi},$$

where $\mathcal{P}$ is the set of sampled pairs and $\xi = 10^{-8}$ ensures numerical stability. A lower distortion indicates that the pairwise distance structure is better preserved after compression.

Semantic Purity. The semantic purity metric (Manning et al., 2008) measures the extent to which each compressed token mixes semantically distinct original tokens. We first obtain semantic pseudo-labels by running k-means on $\mathbf{V}_{\text{orig}}$ with $k = 20$ clusters under a fixed seed, assigning each original token $\mathbf{v}_n$ a label $\ell_n \in \{1, \ldots, k\}$. For all methods, we use the most frequent pseudo-label within each compression group as its empirical reference category for purity evaluation, breaking ties by the smallest label index. The resulting purity measures the fraction of tokens belonging to the group's dominant category. For BGStream, each group corresponds to a non-empty grid cell. We aggregate the group-level purity scores with weights proportional to their token counts, giving equal weight to each source token. This common evaluation rule enables comparison of within-group semantic consistency across methods at matched retention ratios.

We compute each group's purity as the fraction of tokens belonging to its empirical reference category:

$$\text{purity}(c) = \frac{1}{|\mathcal{S}_c|} \max_{\ell\in\{1,\ldots,k\}} \big|\{n \in \mathcal{S}_c : \ell_n = \ell\}\big|,$$

where $S_c = \{n : a(n) = c\}$. We report token-weighted semantic purity:

$$\text{Semantic Purity}_{\text{token}} = \frac{1}{N} \sum_{c\in\mathcal{C}} |\mathcal{S}_c|\, \text{purity}(c),$$

where $\mathcal{C}$ is the set of non-empty compression groups and $N = \sum_{c\in\mathcal{C}} |\mathcal{S}_c|$. This weighting gives equal importance to each source token, rather than to each compressed token, matching the source-token counting perspective of Appendix B.3. The empirical majority pseudo-label serves as a proxy for the geometric reference category in Appendix B.3. Assuming that the pseudo-labels correspond to the modeled semantic categories, the majority label recovers the geometric reference category whenever that category accounts for more than half of a cell's tokens. Conversely, a disagreement requires at least half of the tokens to be contaminants. Under the working model of Appendix B.3, the micro-level failure bound therefore implies that a cell sampled in proportion to its source-token count has a label-disagreement probability of at most $2\mathbb{P}_{\text{micro}}$. Thus, when contamination is limited, as quantified by a small $\mathbb{P}_{\text{micro}}$, the majority label recovers the geometric reference category with high probability in this token-weighted sense, supporting our use of token-weighted majority purity as an empirical measure of semantic consistency. A higher value indicates that a larger fraction of source tokens belongs to the dominant category of its assigned group. For all methods, the pseudo-labels are computed on the same original tokens with an identical seed to ensure a fair comparison.

D.8 Experiments on Offline Long Video Understanding Benchmarks

While BGStream is primarily designed for streaming video understanding, we also evaluate it on three offline long video benchmarks (EgoSchema (Mangalam et al., 2023), LongVideoBench (Wu et al., 2024), and MLVU (Zhou et al., 2025)) to test its generality beyond streaming. As shown in Table 12, we observe that BGStream remains competitive, trailing the leading compression baselines by a modest margin while staying ahead of the rest. This gap relates to a structural difference between two settings. More precisely, strong S+T baselines such as FlashVid (Fan et al., 2026) and Holitom (Shao et al., 2025) exploit **access to the entire video at compression time**: they partition the video by its global structure and suppress temporal redundancy within each segment. The role of this global access is evident in StreamingTOM (Chen et al., 2025): it is also an S+T method, but is designed for streaming and therefore lacks such access. Without it, StreamingTOM performs substantially worse than the offline-designed S+T baselines. The gap between the top S+T baselines and BGStream thus stems from their reliance on a global video prior, an assumption that is unavailable in the streaming regime BGStream targets.

Table 12: Results on offline long video understanding benchmarks. OOM indicates the out-of-memory.

T	Method	EgoSchema	LongVideoBench	MLVU
S	VisionZip (Yang et al., 2025a)	55.60	45.25	56.53
	FastVid (Shen et al., 2025)	58.80	OOM	59.86
S+T	Holitom (Shao et al., 2025)	**64.60**	51.08	63.61
	FlashVid (Fan et al., 2026)	63.40	**51.16**	**64.21**
	StreamingTom (Chen et al., 2025)	60.06	44.43	60.23
	BGStream	62.00	51.08	62.50

D.9 Effect of FIFO Buffer Size

We study how the FIFO buffer size n_r affects accuracy on StreamingBench (Figure 6). With only n_r=1, accuracy drops sharply to 65.16%, indicating that a single recent frame is insufficient to provide the query-relevant perceptual evidence needed near the query timestamp. From n_r=2 onward, accuracy improves steadily, rising from 67.84% to 68.60% as n_r grows from 2 to 5. Since the FIFO term in the retention ratio (Eq. 12) scales linearly with n_r, further enlarging the buffer would inflate the token budget without proportional accuracy gains. We therefore adopt n_r=5, which provides sufficient recent perceptual evidence under our token budget.

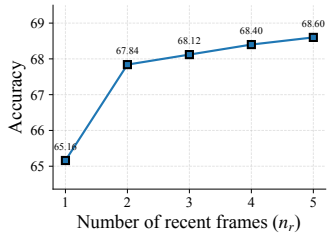


Figure 6: Performance over different FIFO buffer size (n_r).

D.10 Evaluation on Contextual Understanding and Forward Active Responding

While our main quantitative evaluation in Sec. 5 primarily focuses on real-time visual perception and backward tracing, streaming video understanding in deployed environments often demands broader contextual reasoning and proactive responding capabilities. To verify that BGStream's advantages extend beyond real-time perception, we expand our evaluation to the remaining splits of both benchmarks: (i) the *Contextual Understanding* split of StreamingBench (Lin et al., 2024), which covers Multiple-Choice Contextual Understanding (MCU), Open-Ended Contextual Understanding (ACU), Sequential Question Answering (SQA), and Proactive Responding (PO); and (ii) the *Forward Active Responding* split of OVO-Bench (Niu et al., 2025), comprising Real-time Event Categorization (REC), Streaming Status Recognition (SSR), and Continuous Reaction Recommendation (CRR). Following their official evaluation protocols, we compare BGStream against representative visual token compression baselines (VisionZip (Yang et al., 2025a), StreamingTOM (Chen et al., 2025), and FlashVid (Fan et al., 2026)) under a matched $\sim 5\%$ retention budget using LLaVA-OneVision-7B.

Table 13: Performance on the Contextual Understanding split of StreamingBench (Lin et al., 2024) and the Forward Active Responding split of OVO-Bench (Niu et al., 2025).

Method	StreamingBench (Contextual)					OVO-Bench (Forward)			
	MCU	ACU	SQA	PO	Avg.	REC	SSR	CRR	Avg.
VisionZip (Yang et al., 2025a)	26.00	29.20	27.20	10.89	23.32	21.63	65.66	60.42	49.24
StreamingTOM (Chen et al., 2025)	28.40	28.40	28.00	14.11	24.73	**24.21**	65.66	60.83	50.23
FlashVid (Fan et al., 2026)	25.20	32.00	28.40	13.31	24.73	23.50	66.14	61.25	50.30
BGStream	**34.80**	**40.40**	**36.00**	**18.55**	**32.44**	23.35	**68.04**	**64.17**	**51.85**

As shown in Table 13, BGStream achieves the highest overall accuracy across both splits, demonstrating superior performance over all compression baselines. This indicates that the semantic structure preserved by our bilateral grid splatting is not restricted to real-time perception. By maintaining compact, structure-preserving representations across temporal groups, BGStream effectively retains crucial long-horizon context and proactive visual evidence essential for complex streaming tasks.

D.11 Comparison with Recent KV-Cache Methods

To further support our claim that BGStream is complementary to KV-cache optimization beyond ReKV (Di et al., 2025), we combine BGStream with InfiniPot-V (Kim et al., 2025), a recent memory-bounded KV-cache optimization method. We evaluate accuracy, per-query latency (ms), and peak KV-cache memory usage (GiB) on StreamingBench and OVO-Bench using an NVIDIA H200 GPU, without any retraining or modification to either method.

Table 14: Compatibility analysis combining BGStream with InfiniPot-V (Kim et al., 2025).

Method	StreamingBench			OVO-Bench		
	Acc. (%)	Lat. (ms)	KV (GiB)	Acc. (%)	Lat. (ms)	KV (GiB)
InfiniPot-V (Kim et al., 2025)	**73.64**	11,180	0.28	**57.66**	6,636	0.30
InfiniPot-V + BGStream	68.04	5,144	**0.03**	53.90	3,574	**0.03**
BGStream (alone)	68.60	**815**	–	54.58	**1,037**	–

As shown in Table 14, combining InfiniPot-V with BGStream substantially improves computational efficiency while maintaining high accuracy. This suggests that BGStream fundamentally alleviates the compute bottleneck by filtering out visual redundancy before the LLM. In summary, BGStream integrates seamlessly with other KV-cache methods, jointly enabling memory and compute optimization for streaming deployment.

D.12 Ablation Study

Decoupling the FIFO buffer. BGStream is developed by concatenating compressed bilateral grid tokens with raw visual tokens from a recent FIFO buffer. Since recent work (Shen et al., 2026) points out that streaming benchmarks often exhibit a strong recency bias, we perform an ablation study to decouple the contributions of the bilateral grid and the FIFO buffer under a matched token budget constraint. We compare three variants: (i) *Grid-only*, which removes the FIFO buffer and relies solely on grid tokens; (ii) *FIFO-only (matched budget)*, which removes the bilateral grid and increases the per-frame selection ratio so that its total token budget matches BGStream; and (iii) *BGStream (full composition)*. As shown in Table 15, *FIFO-only* alone is still comparable with BGStream, confirming that recent context carries a significant portion of performance on these benchmarks. Nevertheless, under the same token budget, BGStream outperforms *FIFO-only* on both benchmarks. This demonstrates a clear division of labor: while the FIFO buffer provides short-term evidence, allocating a portion of the token budget to the bilateral grid supplies complementary long-horizon context that additional recent frames cannot provide, which is supported by the offline video understanding in Sec. D.8.

Table 15: Ablation study decoupling the bilateral grid and the FIFO buffer under a matched token budget.

Variant	StreamingBench	OVO-Bench
Grid-only	60.88	48.43
FIFO-only	67.96	53.76
BGStream (Full)	**68.60**	**54.58**

Alternative projection methods. To study the effectiveness of Gaussian JL projection of BGStream, discussed in Sec. 4.1, we compare our Gaussian Johnson-Lindenstrauss (JL) projection against a PCA projection (*PCA-group*) as a natural choice for this projection. For each temporal group of $n_g = 32$ frames, *PCA-group* fits a principal component analysis on the visual features once the group completes and projects them into $d_s = 4$ dimensions prior to grid splatting. As shown in Table 16, we observe that *PCA-group* is comparable to Gaussian JL projection, while *PCA-group* requires significant projection time due to per-group singular value decomposition (SVD). This indicates that the core compression advantage stems from the bilateral grid aggregation structure, while Gaussian JL projection remains the optimal choice for streaming deployment, high runtime efficiency, and formal theoretical guarantees.

Table 16: Comparison between Gaussian JL projection and PCA-group projection in BGStream.

Projection	StreamingBench	Proj. Time (ms)
PCA-group	68.36	687.69
Gaussian JL (Ours)	**68.60**	**10.17**

D.13 Normalization Scope in Streaming Setting

Table 17: Effect of min-max normalization scope on StreamingBench and OVO-Bench.

Normalization Scope	StreamingBench	OVO-Bench
Per group ($n_g = 32$)	68.60	54.58
Per frame (on-the-fly)	68.72	54.31

In Sec. 4.2 of the main paper, BGStream applies min-max normalization to guidance keys before grid splatting. To clarify how this normalization operates in a strict streaming setting, we investigate whether group-level statistics introduce lookahead or cell reassignment issues. In our default setup (*Per temporal group*), min-max normalization is computed over each $n_g = 32$ frame group ($N = n_g \cdot n_v$ tokens). This involves buffering a small, constant-size temporal group before splatting; cell assignments are computed once and never revisited. To evaluate BGStream under the strictest interpretation with zero lookahead, we test a *Per-frame (fully on-the-fly)* variant, where min-max statistics are computed independently per frame, allowing each frame to be splatted into the grid immediately upon arrival. As shown in Table 17, we observe that the per-frame variant is on par with the per-group setup, indicating that bilateral grid assignment is highly robust to the normalization scope and that BGStream operates reliably even under zero-lookahead, frame-by-frame streaming conditions.

D.14 Analysis of Computational Cost

To assess the practical efficiency of BGStream beyond end-to-end latency, we analyze the computational overhead introduced by our compression mechanism relative to the full Video-MLLM pipeline. Using LLaVA-OneVision-7B (Li et al., 2024) on a single NVIDIA H200 GPU, we decompose the per-query inference latency on StreamingBench into three distinct stages: 1) *Visual Encoder + Adaptor*, which encodes incoming video frames into visual features; 2) *BGStream Compression*, which performs Gaussian JL key projection, normalization, and bilateral grid splatting; and 3) *LLM-side Processing*, which performs self-attention prefill and decoding over the compressed visual context. As broken down in Table 18, BGStream's compression mechanism consumes only a small amount of latency compared to the 1) and 3). The dominant remaining cost resides in the vision encoder (64.3%), an orthogonal front-end component shared by all Video-MLLM architectures, confirming that BGStream provides a highly efficient compression module for streaming inference.

Table 18: Stage-wise per-query latency breakdown on StreamingBench (LLaVA-OneVision-7B).

Pipeline Stage	Latency (ms)	Ratio (%)
1) Visual Encoder + Adaptor	524.2	64.3%
2) BGStream Compression	95.4	11.7%
3) LLM (Prefill + Decode)	195.4	24.0%
Total End-to-End Latency	815.0	100.0%