LLaVA-OneVision-2: Towards Next-Generation Perceptual Intelligence

LMMs-Lab Glint Lab, AIM for Health Lab, MVP Lab

We introduce **LLaVA-OneVision-2 (LLaVA-OV-2)**, the most capable vision-language model in the LLaVA-OneVision series to date, achieving superior performance across a broad range of multimodal benchmarks. The model builds on a native OneVision-Encoder and incorporates Windowed Attention for efficient local computation while maintaining native resolution. Its key advance is **codec-stream tokenization**: it treats compressed video as a continuous bit-cost stream, where bit-cost dynamics determine adaptive temporal groups, and motion-residual cues select salient spatial evidence into compact visual canvases. This allocation concentrates a limited token budget on event-bearing content, enabling more stable long-video token compression than fixed groups of pictures. A shared 3D RoPE further places codec canvases, sampled frames, and images in a unified spatiotemporal coordinate system. Furthermore, we build the LLaVA-OV-2 data and training stack around large-scale open supervision: approximately 8M re-captioned video samples for pretraining, a 4M-sample spatial corpus for fine-tuning. We also introduce **JumpScore**, a temporal-localization benchmark targeting fine-grained grounding in high-frequency, densely repeated motion, a regime underrepresented by existing video evaluations. A standout capability of LLaVA-OV-2 is its unified perception across video understanding, temporal grounding, spatial grounding, and manipulation-trace reasoning. On JumpScore, LLaVA-OneVision-2-8B reaches 74.9 JumpScore mAP, surpassing Qwen3-VL-8B (30.1) by +44.8 points; under matched visual-token budgets on the same benchmark, codec-stream inputs improve temporal grounding over frame sampling by +9.7 points. Across standard benchmarks, LLaVA-OneVision-2-8B further outperforms Qwen3-VL-8B by +4.3 average points on video tasks, +5.3 on spatial tasks, and +15.6 average J&F on tracking tasks. Our code, data, and models are released as open-source resources.

Date: May 24, 2026
Code: https://github.com/EvolvingLMMs-Lab/LLaVA-OneVision-2
Data: https://huggingface.co/datasets/mvp-lab/LLaVA-OneVision-2-Data
Model: https://huggingface.co/lmms-lab-encoder/LLaVA-OneVision-2-8B-Instruct

1 Introduction

Recent open Large Vision-Language Models (LVLMs) (Bai et al., 2025a,b; Zhu et al., 2025; An et al., 2025; Yang et al., 2025a,c; Zhang et al., 2026a; Clark et al., 2026; Liu et al., 2024b; Zhang et al., 2025a; Zohar et al., 2024; Wang et al., 2025c; Liu et al., 2024a; Shen et al., 2024) largely retain a frame-centric observation paradigm: *Uniform frame sampling or Mixed-resolution frames*, which combines sparse high-resolution key frames with denser low-resolution context frames to satisfy a fixed token budget. Yet such designs still reduce video to a set of decoded frames, underrepresenting continuous spatial structure and motion dynamics while overlooking the predictive stream signals that make video uniquely informative. Video codecs such as H.264 and H.265/HEVC *(High Efficiency Video Coding)* decompose video signals into spatially complete intra-coded frames (I-frames) that establish global context and predicted frames (P-frames) that encode inter-frame variations via motion compensation and residuals (Sullivan et al., 2012). The OneVision-Encoder (OV-Encoder) (Tang et al., 2026) is an early prototype along this path: it introduced codec patchification as a backbone-side primitive and showed that, under a fixed token budget, codec-selected I/P patches provide the language model with denser discriminative evidence than uniformly sampled frame patches.

In this paper, we argue that next-generation perceptual intelligence should move beyond uniformly observing frames toward selectively allocating evidence in predictive visual streams, where most pixels sustain contextual

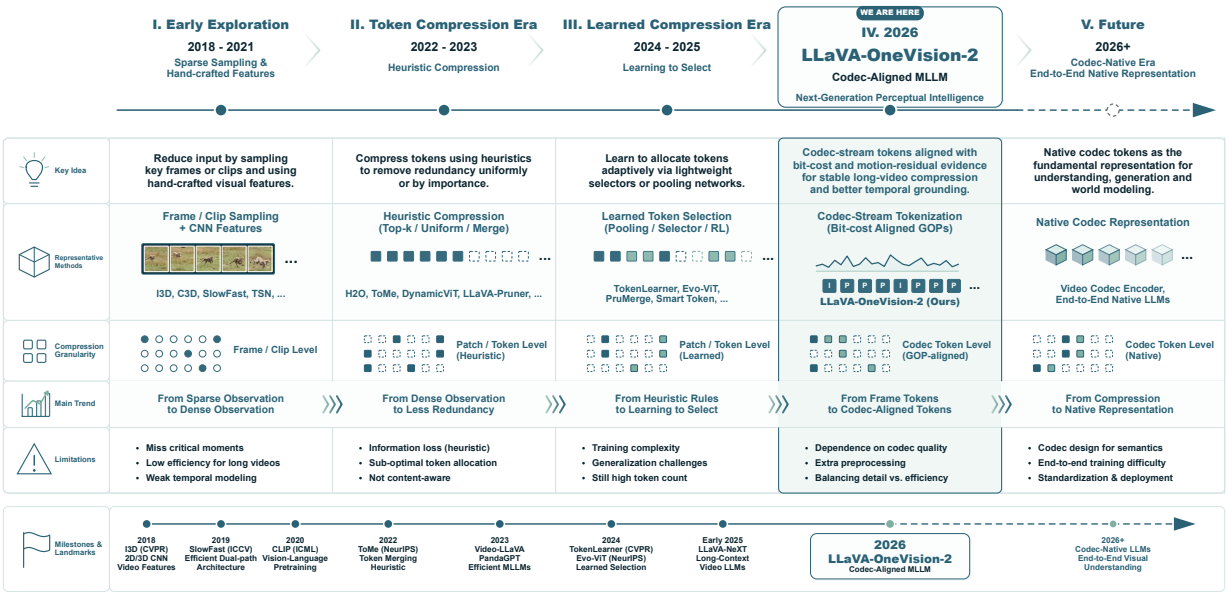


Figure 1 Roadmap of video understanding from token compression to codec-aligned perceptual intelligence. The roadmap traces the evolution from early frame/clip sampling and hand-crafted visual features, to heuristic token compression, learned token selection, and the 2026 codec-aligned paradigm represented by LLaVA-OneVision-2.

continuity and only sparse deviations encode discriminative semantic, spatial, and temporal structure.

We introduce **LLaVA-OneVision-2**, the most capable vision-language model in the LLaVA-OneVision series to date, achieving strong performance across a broad range of multimodal benchmarks. The model builds on a native dynamic-resolution OV-Encoder with codec patchification, and augments it with a codec-adaptive attention interface that combines spatial windowed attention for efficient local computation with group-visible masks while preserving native resolution. Its key advance is **codec-stream tokenization**: It treats compressed video as a continuous bit-cost stream. Bit-cost dynamics adaptively determine temporal group boundaries, while motion-residual cues condense salient spatial evidence into compact, merge-aligned visual canvases. Rather than allocating visual tokens by elapsed time or fixed frame slots, this stream-aware design makes token density follow the evolving bit-cost-residual profile of the compressed stream, densifying around perceptual transitions while thinning over predictable intervals, thereby enabling more stable long-video token compression than fixed Group of Pictures (GOPs). A shared 3D RoPE further places codec canvases, sampled frames, and images in a unified spatiotemporal coordinate system.

LLaVA-OneVision-2 is trained with a progressive four-stage recipe that scales supervision from image grounding to long-video and spatial reasoning. Stage 1 mixes $\sim$85M image–text samples with 4.2M 30s video captions at 30 frames; Stage 2 adds large-scale instruction data ($\sim$22M LLaVA-OneVision-1.5 and $\sim$24M FineVision samples) together with 2.7M 30–60s and 700K 60–180s video captions at 60/90 frames; Stage 3 extends to long-form video with video-instruction corpora and 350K 10–15min captions at 384 frames; and Stage 4 re-encodes long videos through the variable-length-GOP codec pipeline at 384/768 frames while adding LLaVA-OneVision-2-Spatial-4M for 2D/3D spatial supervision. Across all stages, codec-patchified videos, uniformly sampled videos, and image/tiled inputs are interleaved in training.

Existing video benchmarks underrepresent fine-grained grounding in high-frequency, densely repeated motion, where the challenge is not merely recognizing the event category but localizing the correct action instance among many visually similar cycles. We therefore introduce JumpScore, a temporal-localization benchmark designed to evaluate perceptual transition-level grounding. On JumpScore, LLaVA-OneVision-2-8B achieves 74.9 JumpScore mAP, substantially outperforming Qwen3-VL-8B (30.1) by +44.8 points. Within the same benchmark and under matched visual-token budgets, codec-stream inputs improve temporal grounding over frame-sampling inputs by +9.7 points. At the model level, LLaVA-OneVision-2-8B outperforms Qwen3-VL-8B by +4.3 points on average across video benchmarks, +5.3 points across spatial benchmarks, and +15.6 average

J&F on tracking benchmarks. Our experiments have revealed: codec-stream inputs favor long-video tasks governed by coarse temporal structure, such as temporal grounding, event understanding, event ordering, and salient retrieval, by reallocating tokens to high-bit-cost intervals and high-residual regions. In contrast, frame sampling remains preferable for detail-sensitive queries, where decisive cues are static, fine-grained, spatially small, trajectory-specific, or boundary-level, because dense frame observations better preserve local texture, subtle appearance cues, and frame-to-frame continuity.

In summary, the main contributions of LLaVA-OneVision-2 are as follows:

1. LLaVA-OneVision-2 is a codec-aligned MLLM whose codec-stream tokenization treats video as a continuous bit-cost stream, aligning visual-token allocation with bit-cost dynamics and motion-residual evidence to enable stable long-video token compression.
2. We scale training with approximately 8M re-captioned video samples and a 4M-sample 2D/3D spatial corpus, and introduce JumpScore, a temporal-localization benchmark for fine-grained video grounding in high-frequency, dense motion. Our code, data, and models are released.
3. LLaVA-OV-2-8B delivers consistent gains over Qwen3-VL-8B, improving the average score by +4.3 points across 18 video tasks, +5.3 points across 11 spatial-reasoning tasks, and +15.6 average J&F across 4 tracking tasks. Codec-stream inputs further improve temporal grounding over frame sampling by +9.7 points, and LLaVA-OV-2-8B reaches 74.9 JumpScore mAP against Qwen3-VL-8B's 30.1 (+44.8 points).

2 Architecture

This section describes the model-side design of LLaVA-OneVision-2, as illustrated in Figure 2. §2.1 first gives the full multimodal stack, consisting of a Vision Encoder, a lightweight vision-language connector, and an autoregressive language model decoder. §2.2 then focuses on the visual encoder interface: how sampled frames, codec-patchified videos, and static images are represented as visual canvases with token metadata, and group-visible attention masks.

2.1 LLaVA-OneVision-2

Vision Encoder. LLaVA-OneVision-2 adopts the OneVision-Encoder (Tang et al., 2026) as a shared backbone for sampled-frame videos, codec-stream videos, and static images, mapping all inputs into a unified visual-token interface with patch embeddings, 3D positional coordinates, and encoder-side group assignments. Shared 3D RoPE provides a common spatiotemporal coordinate system, while group-visible masks define token visibility: sampled-frame and IPPP-style inputs use fixed four-slot groups, static images use a degenerate single-temporal group, and codec-stream inputs use bit-cost-adaptive GOP ids to group tokens from the same variable-length GOP across P-canvases. Following native-resolution vision-transformer designs (Dehghani et al., 2023; Beyer et al., 2023; Tschannen et al., 2025; Bai et al., 2025b), spatial windowed attention is used in most visual layers for efficient native-resolution processing and remains orthogonal to the video-level grouping rule.

Vision–Language Connector. A lightweight two-layer MLP maps OneVision-Encoder representations into the language-model embedding space. Because sampled-frame videos, IPPP-style windows, codec-derived I/P canvases, and static images share the same encoder-output format, the connector remains interface-invariant across input forms. Codec-stream processing therefore changes only the evidential structure presented to the visual encoder, while leaving the vision-language alignment interface unchanged.

Large Language Model. The projected visual tokens are paired with the text instruction and decoded by a shared Qwen3-8B autoregressive language model under the supervised next-token objective. No codec-specific adapter, reconstruction decoder, or language-side branch is introduced. Consequently, frame-sampled and codec-stream inputs differ only in evidence selection and attention-group assignment, whereas the encoder–connector–decoder pathway remains architecturally identical.

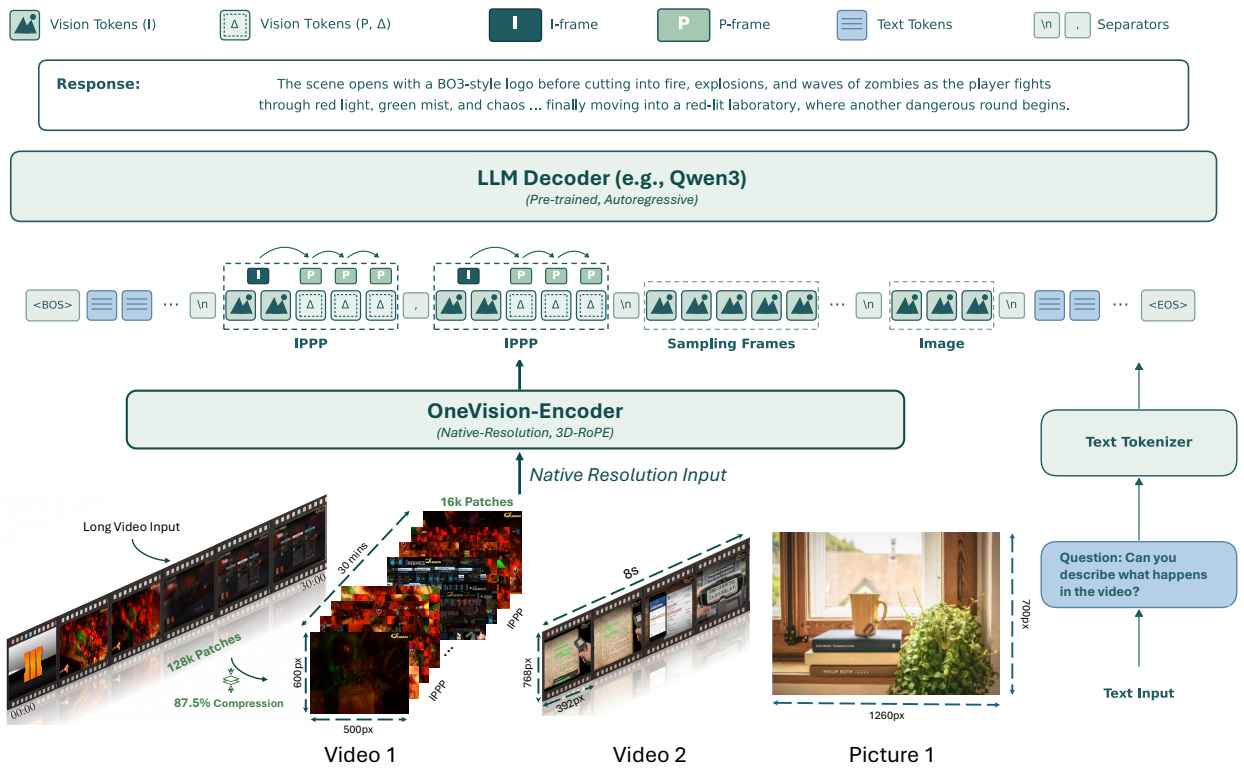


Figure 2 LLaVA-OneVision-2 architecture. The model unifies codec-stream videos, sampled-frame videos, and native-resolution images under a shared visual-token interface. Codec inputs are encoded as I/P visual canvases, sampled videos as frame-token sequences, and images as spatial visual tokens; all inputs are processed by the OneVision-Encoder. The resulting visual embeddings are combined with text tokens and decoded by a pre-trained autoregressive language model, allowing a single architecture to support video and image understanding.

2.2 Codec-stream Tokenization

Unified Visual-token Interface. For a video $\mathcal{V}$, the codec front-end emits visual canvases, token metadata, and adaptive temporal groups:

$$\mathfrak{C}(\mathcal{V}) = (\mathcal{X}, \mathcal{U}, \mathcal{G}), \qquad \mathcal{X} = \{(X_s, \ell_s)\}_{s=0}^{S-1},\ \ell_s \in \{\mathrm{I}, \mathrm{P}\}, \qquad \mathcal{U} = \{u = (\iota_u, f_u, \mathbf{p}_u^{\mathrm{can}}, \mathbf{p}_u^{\mathrm{src}}, \kappa_u)\}_{u=1}^{N}. \tag{1}$$

Here $\mathcal{X}$ contains S I/P canvases, $\mathcal{U}$ contains N visual-token records, and $\mathcal{G} = \{\mathcal{G}_k\}$ denotes the induced codec groups. For token u, ι_u is the canvas index, f_u is the source-frame id, $\mathbf{p}_u^{\mathrm{can}}$ is the packed canvas coordinate, $\mathbf{p}_u^{\mathrm{src}}$ is the source-frame patch coordinate, and κ_u is the bit-cost-adaptive group id. The packed coordinate supports compact canvas construction, while the source coordinate preserves the spatial origin of each token for spatiotemporal encoding. The connector and language model do not consume these codec fields directly; codec-stream tokenization affects the model by selecting visual evidence and assigning token visibility groups.

Groups of Pictures (GOPs) Partition. Rather than assigning visual slots by elapsed time, codec-stream tokenization partitions video according to the temporal bit-cost profile of the compressed stream. We divide the video into B bins of duration Δ and aggregate the packet size of predicted frames within each bin:

$$e_b = \sum_{q \in \mathcal{P}_{\mathrm{PB}}} \mathrm{bytes}(q)\, \mathbf{1}\{\tau(q) \in [b\Delta, (b+1)\Delta)\}, \qquad \theta = \frac{\sum_{b=0}^{B-1} e_b}{\max(1, K_{\mathrm{tar}})}. \tag{2}$$

Here $\mathcal{P}_{\mathrm{PB}}$ is the set of P/B-frame packets, $\mathrm{bytes}(q)$ is the packet size used as a proxy for prediction bit-cost, $\tau(q)$ is the presentation timestamp of packet q, and K_{tar} is the target number of temporal groups. Thus, e_b is a bin-level bit-cost rather than a per-frame score, and θ is the average P/B bit-cost quota per adaptive GOP.

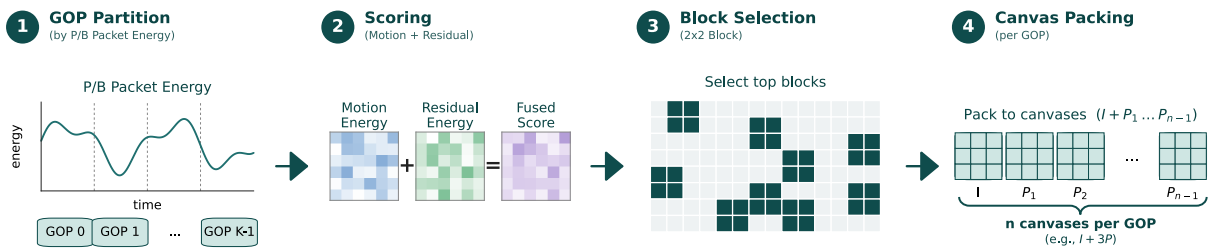


Figure 3 Codec-stream tokenization. P/B packet bit-cost partitions the video into adaptive GOPs; motion and residual signals jointly score spatial saliency; high-score 2×2 patch blocks are selected and packed into compact I/P canvases. Each GOP yields one anchor I-canvas and multiple P-canvases carrying motion-residual evidence, producing merge-aligned visual tokens whose density follows the bit-cost–residual profile of the stream rather than fixed frame slots.

I-frame packets are excluded because they mainly reflect intra-frame spatial complexity, whereas P/B packets expose inter-frame prediction difficulty, motion, and residual change.

Starting from bin s_k, the next boundary is triggered when the current segment either reaches the maximum span or accumulates sufficient bit-cost after the minimum span:

$$i_k = \min \left\{ i \geq s_k : (i - s_k + 1 \geq L_{\max}) \vee \left[i - s_k + 1 \geq L_{\min} \wedge \sum_{b=s_k}^{i} e_b \geq \theta \right] \right\}, \tag{3}$$

where $L_{\min} = \lceil T_{\min}/\Delta \rceil$ and $L_{\max} = \lceil T_{\max}/\Delta \rceil$ are the minimum and maximum group spans measured in bins. The tentative boundary i_k is then refined by local valley search:

$$c_k = \underset{b \in \mathcal{N}_k(i_k)}{\overset{\text{lex}}{\arg\min}} (e_b, |b - i_k|), \qquad \mathcal{G}_k = [s_k, c_k], \qquad s_{k+1} = c_k + 1. \tag{4}$$

The search window $\mathcal{N}_k(i_k)$ is centered around i_k and constrained by the minimum span, maximum span, and video endpoints. The lexicographic rule first selects the lowest-bit-cost valley and then chooses the closest bin to the trigger point. High-change intervals therefore reach the quota quickly and form shorter groups, while predictable intervals span longer groups. A token u receives $\kappa_u = k$ if its source-frame time f_u/fps falls inside $[s_k\Delta, (c_k + 1)\Delta)$. Figure 4 visualizes this bit-cost-based adaptive grouping process.

Scoring and Block Selection. Within each bit-cost-adaptive group, motion-residual evidence determines which spatial regions are preserved. For a predicted frame t, the codec exposes motion vectors $\mathbf{d}_t$ and a luma residual r_t^Y. As in the OV-Encoder, the motion field is densified to the pixel grid, the residual is interpreted around its zero point 128, and the two signals are normalized by robust percentile statistics. This gives a dense saliency map $S_t(\mathbf{x}) = \overline{M}_t(\mathbf{x}) + \overline{R}_t(\mathbf{x})$, where $\overline{R}_t$ denotes the normalized residual-response map, obtained by measuring the absolute luma-residual deviation from the zero point 128, scaling it by a robust frame-level percentile, and clipping it to $[0, 1]$. Likewise, $\overline{M}_t$ is the percentile-normalized motion-magnitude map derived from the densified codec motion vectors.

The difference from the original OV-Encoder patch mask is the selection granularity. Instead of selecting individual high-score patches, the codec patch-GOP path aggregates saliency into 2×2 patch blocks. Let $\mathcal{P}_{h,w}$ denote the $p \times p$ region of patch (h, w), with $p = 16$ in our implementation. The block score is $A_{t,i,j} = \sum_{\alpha,\beta \in \{0,1\}} \sum_{\mathbf{x} \in \mathcal{P}_{2i+\alpha,\, 2j+\beta}} S_t(\mathbf{x})$. Thus, a selected unit always contains the four neighboring patches $(2i, 2j)$, $(2i, 2j+1)$, $(2i+1, 2j)$, and $(2i+1, 2j+1)$. This block-level primitive is aligned with the encoder-side 2×2 merge operation: every selected block contributes four spatially coherent patch tokens, avoiding the downstream merging of unrelated patches from different source regions or frames. We further augment the motion-residual score with a normalized patch-level bit-cost prior. Since bit-cost is naturally available at block granularity, it is fused during 2×2 block scoring rather than projected back to the pixel grid. The bit-cost term reflects local coding complexity and complements motion and residual energy for codec-aware spatial token allocation.

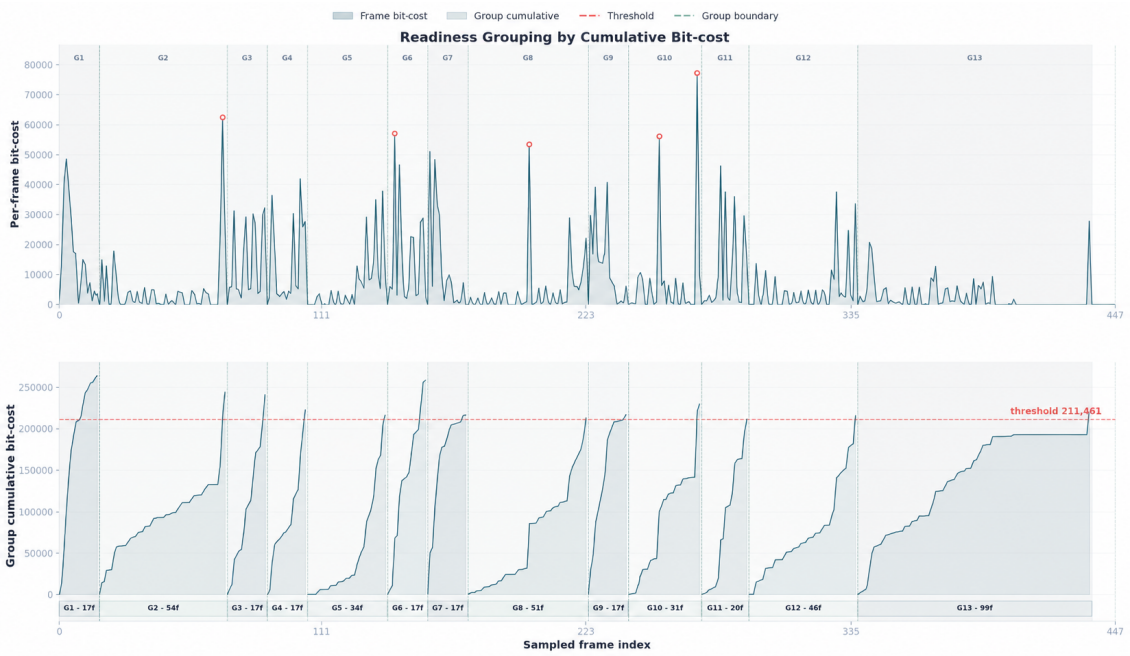


Figure 4 Codec-stream grouping by cumulative bit-cost. For example, 448 sampled frames are divided into 13 codec-stream groups under a cumulative bit-cost threshold of 211,461. The top panel shows the frame-level bit-cost contribution in blue, where sharp peaks typically indicate rapid motion, viewpoint changes, or abrupt visual transitions. The bottom panel shows the cumulative bit-cost within each group in orange, which resets after every group boundary and approaches the red threshold before a new group is opened. Green dashed lines mark the resulting codec-stream group boundaries, and the bottom color bands indicate the number of frames covered by each group.

Canvas Packing. A global top-ranked selection over an entire codec group can over-concentrate tokens on a single high-response frame. We therefore construct P-canvases through stratified temporal allocation. Let $\mathcal{Z}_k = \{z = (t, i, j, A_{t,i,j})\}$ be the candidate 2×2 patch blocks inside group k, where t is the source frame, (i, j) is the block coordinate, and $A_{t,i,j}$ is the block saliency score. For each frame t, we sort its candidate blocks by $A_{t,i,j}$ in descending order and denote by $\rho_t(z)$ the zero-based rank of candidate z within that frame. We then attenuate repeated candidates from the same frame:

$$\widetilde{A}_z = \frac{A_{t,i,j}}{\sqrt{1 + \lambda \rho_t(z)}}, \qquad w_t = \sum_{z \in \mathcal{C}_k(t)} \max(0, \widetilde{A}_z) + \alpha_{\text{peak}} \max_{z \in \mathcal{C}_k(t)} \widetilde{A}_z. \tag{5}$$

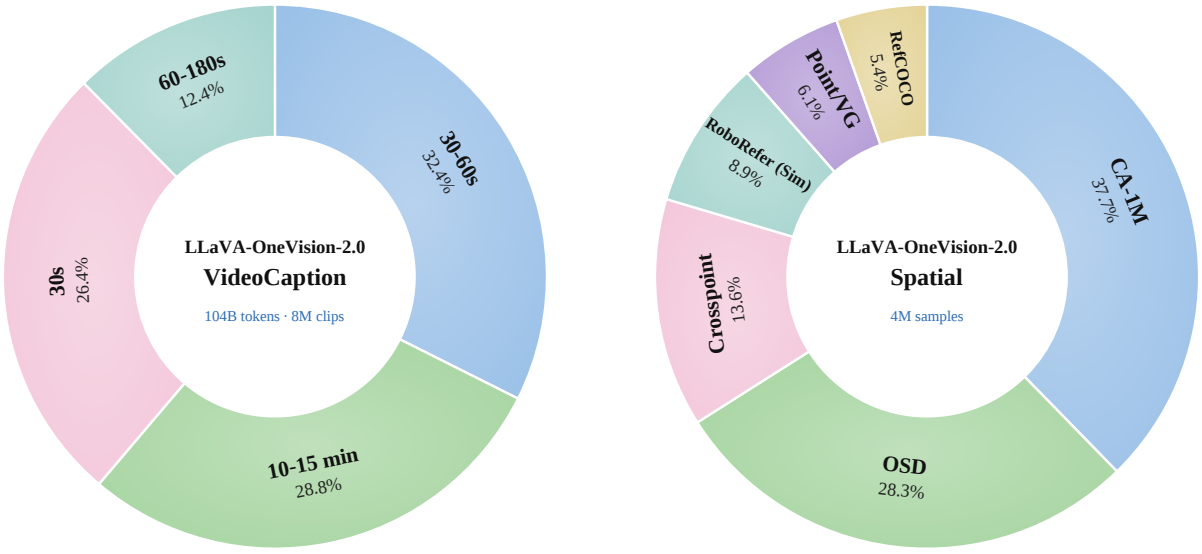
Here $\mathcal{C}_k(t) \subset \mathcal{Z}_k$ is the set of candidate blocks from frame t within group k, λ controls the strength of same-frame attenuation, α_{peak} weights the strongest frame-level response, and w_t is the resulting frame-level allocation mass. The attenuation prevents a single high-response frame from dominating the entire group, while the peak term preserves frames that contain a highly localized but important response.

To assign P-canvases across time, we sort candidate frames in group k as $\{t_n\}_{n=0}^{M_k-1}$ and compute the cumulative allocation curve

$$F_k(\ell) = \frac{\sum_{n=0}^{\ell} w_{t_n}}{\sum_{n=0}^{M_k-1} w_{t_n}}, \qquad 0 \le \ell < M_k. \tag{6}$$

This curve maps the temporal order within group k to the fraction of accumulated saliency mass. If group k is assigned m_k P-canvases, the r-th P-canvas, $r \in \{0, \ldots, m_k - 1\}$, draws high-scoring non-duplicate blocks from the frames whose cumulative allocation mass falls in $[r/m_k, (r+1)/m_k)$. When the corresponding interval contains too few candidates, the selector expands to neighboring frames and finally falls back to the full group. Thus, bit-cost dynamics determine where temporal resolution is needed, while motion-residual saliency and frame-level allocation weights determine how each group is covered by P-canvases.

Group-visible Attention. Codec-stream inputs, sampled-frame inputs, and static images share the same patch



(a) Video-Caption Data Mixture **(b)** Spatial Data Mixture

Figure 5 LLaVA-OneVision-2 data mixtures. (a) Token-volume proportions of the video-caption corpus and (b) the spatial-reasoning corpus used during training. The video-caption mixture contains 104.1B tokens from 7.96M clips spanning four duration buckets, while the spatial mixture aggregates 4M samples drawn from six datasets covering 3D scenes, spatial reasoning, pointing, and referring expressions.

embedding and 3D positional encoding, forming a unified spatiotemporal token space. The OneVision-Encoder then uses a non-causal group-visible attention interface to define token visibility: sampled-frame and IPPP-style inputs use fixed four-slot groups, codec-stream inputs use the bit-cost-adaptive GOP id κ_u so tokens from the same variable-length GOP remain group-visible across P-canvases, and static images reduce to a single-temporal group. Consequently, all input forms share the same encoder parameters, with only evidence allocation and group assignment varying across inputs.

3 Training Data

The LLaVA-OneVision-2 recipe consumes data from three buckets, each contributing a distinct slice of the model's eventual capability surface.

3.1 Inherited Image–Text Foundation

Image–text foundation. We initialise from the image-pretrained checkpoint of LLaVA-OneVision-1.5 and reuse the LLaVA-OneVision-1.5 mid-training and instruction corpora as-is. The mid-training corpus (LLaVA-OneVision-1.5-Mid-Training-85M) is concept-balanced over ∼85M image–text pairs (20M ZH + 65M EN); the instruction corpus (LLaVA-OneVision-1.5-Instruct-Data, ∼22M samples) covers OCR, GUI, document, grounding, counting, and chart/diagram tasks. We additionally include FineVision (∼24M instruction samples) for broader image-instruction coverage. Detailed per-source statistics for the mid-training and instruction corpora are reported in the LLaVA-OneVision-1.5 release; we do not re-derive or modify them here.

Inherited video instruction. For video instruction tuning, we utilize relevant subsets from four publicly available corpora: LLaVA-Video-178K (Zhang et al., 2024) (1.6M samples covering captioning, open-ended QA, and multiple-choice QA), VideoChat-Flash-Training-Data (Li et al., 2025b), Molmo2 (Allen Institute for AI, 2025; Clark et al., 2026), and TimeLens. Only the data pertinent to our methodology is selected from each corpus. These corpora are general video-instruction data rather than long-form sources, and we deliberately

do not synthesize any additional long-video instruction data: all long-context capability is acquired from the length-stratified caption corpus, while the inherited corpora supply instruction-following diversity.

3.2 Length-Stratified Video Caption Corpus

Length stratification as a design choice. A central component of our training data is a length-stratified video caption corpus spanning 30 seconds to 15 minutes, totalling approximately 8M captioned clips. We deliberately stratify by length because uniform-length captioning recipes (typically dominated by short clips) over-represent semantic perception relative to temporal continuity: the model learns to "describe a scene" but not to "maintain state across ten minutes." The four buckets (30s, 30–60s, 60–180s, 10–15min) are sized so that each successive stage of the training recipe (§4) can extend its frame budget by a factor of 2–4× without out-running its caption supervision. We compute image tokens at 392×392 input with ViT patch size 14 and a 2×2 vision merge, yielding 196 visual tokens per frame; caption tokens are measured with the Qwen3 tokenizer over a 1,500-sample average per bucket and then scaled by row count.

Codec-aware re-encoding at Stage 4. The 10–15-minute bucket is consumed twice in Stage 4 (§4.4): once at 384 frames under the variable-length-GOP, bit-cost-scored codec configuration of §2.2, and once at 768 frames under the same configuration to densify the temporal axis at the upper end of the recipe's frame-budget schedule. No new captions are produced for the densified pass; the same per-clip caption is re-aligned against a denser visible-patch index.

4 Training Recipe

The LLaVA-OneVision-2 recipe runs in four progressive stages (§4.1–§4.4).

4.1 Stage 1 – Bootstrap from LLaVA-OneVision-1.5 + 30s Video Captions

We initialize from the image-pretrained LLaVA-OneVision-1.5 (An et al., 2025) 8B checkpoint and bootstrap it into a video-aware model by mixing in short video-caption data. The training corpus consists of (i) LLaVA-OneVision-1.5-Mid-Training-85M (An et al., 2025), a concept-balanced image–text dataset, and (ii) our newly released **30s-Video-Caption-4.2M** corpus, where each sample is paired with a caption over a 0–30s video span, sampled at 1 fps with up to 30 frames. All video inputs in this stage are constructed with standard frame sampling; codec-stream tokenization is not used.

4.2 Stage 2 – Instruction Tuning + 30–60s Video Captions

Stage 2 introduces large-scale multimodal instruction data and extends video understanding to medium-length clips. The training mixture consists of (i) LLaVA-OneVision-1.5-Instruct-Data (∼22M samples) (An et al., 2025), (ii) HuggingFaceM4/FineVision (∼24M samples), (iii) our newly released **30s–60s-Video-Caption-2.7M** corpus, where each example is paired with a caption over a 30–60s video span sampled at 1 fps with up to 60 frames, and (iv) our newly released **60s–180s-Video-Caption-700K** corpus, where each example is paired with a caption over a 60–180s video span using up to 90 frames. The maximum frame budget is increased from 30 to 90 in this stage, marking the first expansion step in the cross-stage schedule described in §5. All video inputs in this stage continue to use standard frame sampling without codec-stream tokenization.

4.3 Stage 3 – Long Video Understanding

Stage 3 extends training to long-form video understanding by combining long-duration caption data with established video instruction corpora. The training mixture includes LLaVA-OneVision-1.5-Instruct-Data, HuggingFaceM4/FineVision, lmms-lab/LLaVA-Video-178K (Zhang et al., 2024), OpenGVLab/VideoChat-Flash-Training-Data (Li et al., 2025b), and our newly released **10min–15min-Video-Caption-350K** corpus, which pairs captions with 10–15 minute video spans. The maximum frame budget is increased from 90 to 384 in this stage. As in the previous stages, all video inputs are constructed with standard frame sampling.

4.4 Stage 4 – Extended Video and Codec-Stream Training

Stage 4 further extends long-video training and introduces dedicated spatial supervision. Up to Stage 3, the recipe relies predominantly on caption-style supervision, which provides only indirect learning signals for fine-grained spatial structure. In the final stage, codec-stream tokenization is introduced for the **10min–15min-Video-Caption-350K** corpus. Specifically, we revisit this long-video caption corpus with the variable-length-GOP, bit-cost-scored codec pipeline described in §2.2. The corpus is used both at up to 384 frames and in a densified variant with up to 768 frames under the same codec-stream configuration, making it the largest visual-budget caption component in the training recipe. All remaining Stage-4 data, including multimodal instruction-tuning data, spatial QA data, Molmo2-VideoTrack, and Molmo2-VideoPoint, continues to use the standard input format of the corresponding data source and does not apply codec-stream tokenization. Therefore, codec-stream is not a global Stage-4 preprocessing rule; it is a targeted input representation for the long-video caption component.

Stage 4 also incorporates two spatially oriented data sources. Our in-house **LLaVA-OneVision-2-Spatial-4M** corpus provides 4M structured QA pairs covering size, direction, count, distance, and appearance order over annotated indoor scans, embodied-simulator trajectories, and pseudo-annotated web video frames. We further adopt Molmo2-VideoTrack and Molmo2-VideoPoint (Allen Institute for AI, 2025) for point-based tracking and spatio-temporal pointing. Together, these datasets extend the supervision signal from static spatial relations to temporally grounded spatial understanding.

5 Implementation Details

Three cross-stage design choices are shared across all four stages and cannot be ablated by reading any single stage's data list. Three cross-stage design choices govern the LLaVA-OneVision-2 training recipe beyond the data mixture of any individual stage:

Mixed-batch Composition. Each training step interleaves three input forms from the unified OneVision-Encoder interface: codec-stream videos, uniformly sampled videos, and static image inputs. In practice, the mixture is approximately 50% codec-patchified video, 37.5% uniform chunk-wise video, and 12.5% image inputs. This composition exposes the model to both stream-aware event allocation and frame-faithful visual evidence, while preserving image-understanding capability. Because all input forms share the same patch embedding, 3D RoPE, and visible-token interface, the mixture requires no modality-specific routing or additional input-form parameters.

Frame-budget Schedule. The visible per-clip frame budget increases progressively across stages: 30 frames in Stage 1, 60/90 frames in Stage 2, 384 frames in Stage 3, and 384/768 frames in Stage 4. This schedule matches the temporal span of the supervision: short-video captions support early-stage perception, while long-video captions and codec re-encoding support dense long-form observation in later stages. Since training does not exceed 768 visible frames, denser or hour-scale inputs remain an extrapolation regime for future streaming and ultra-long-context codec modeling.

Cross-stage Codec Scheduling. Stages 1–3 use standard frame-sampling inputs and do not apply codec-stream tokenization. Codec-stream training is introduced in Stage 4, where we switch the long-video data to the variable-length-GOP, bit-cost-scored codec pipeline described in §2.2. The 10–15 minute video-caption corpus is re-encoded under this Stage-4 configuration at 384 and 768 frames. Because codec-stream is enabled only in the final stage, we evaluate codec-stream versus uniform frame sampling under the Stage-4 setting, where the two input strategies share the same model, supervision data, training schedule, and evaluation protocol.

6 JumpScore

Standard temporal-grounding benchmarks such as Charades-STA (Gao et al., 2017), ActivityNet (Caba Heilbron et al., 2015), and QVHighlights (Lei et al., 2021) target one-shot event localization, where adjacent visual evidence is already distinguishable. **JumpScore** probes the opposite regime: fine-grained grounding in

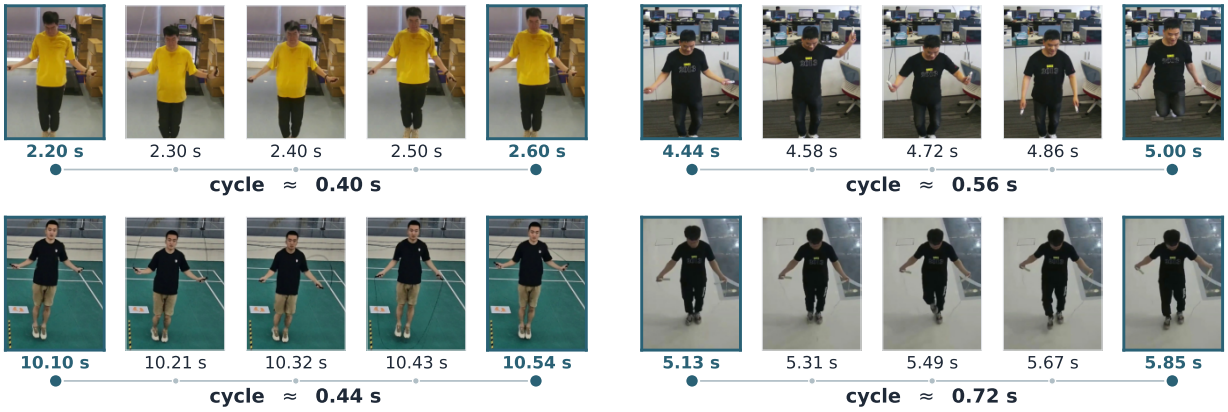


Figure 6 Representative cycles from the JumpScore benchmark. Four clips, each decomposed into five frames spanning one jump-rope cycle. The first and last frame of every panel are ground-truth cycle starts (rope behind legs). The four panels span warehouse, office, sports-court, and tiled-corridor captures.

high-frequency, densely repeating motion where adjacent cycles are visually near-identical and the discriminative signal lives at cycle boundaries. The benchmark consists of 189 in-the-wild jump-rope videos with complete decimal-second annotations of every cycle start, defined by the moment the rope passes behind the legs. Figure 6 shows four representative clips spanning the range of scenes, camera angles, and cycle periods in the benchmark. The dataset is publicly released on Hugging Face[1] and is integrated into the `lmms-eval-ov2` evaluator used elsewhere in this paper.

Dataset Construction. The 189 source videos span multiple indoor scenes, camera angles, and capture devices, with clip durations concentrated in the 30–90 s range so the typical clip contains tens of cycles at sub-second spacing. Capture resolution is at least 1280×720 on every clip and 1920×1080 or higher on more than 84% of them, keeping the rope-and-leg cue visually resolvable. Cycle-start timestamps are annotated at decimal-second precision and verified against the video's source frame rate so that the same boundary frame is recovered under deterministic decoding. Targeting a single, visually well-defined motion primitive isolates cycle-boundary localization from confounders such as category recognition, scene parsing, and text-and-dialogue cues, which is what makes sub-second annotation tractable across the full 189-clip set.

Task and Metric. Each video is paired with a fixed natural-language prompt: "List the start timestamps in s of each jump rope the main character does in the video. The start is defined as the moment the rope is behind the legs." The expected output is a list of decimal-precision timestamps in seconds; no auxiliary modality is provided, and the same prompt is used for every video so that performance differences reflect grounding capability rather than prompt sensitivity. Predictions are matched greedily to ground-truth timestamps within a temporal tolerance band of δ seconds: a predicted timestamp counts as a true positive if it falls within δ of an unmatched ground-truth timestamp, remaining predictions count as false positives, and unmatched ground-truth timestamps count as false negatives. The headline metric is mean Average Precision (mAP) averaged across three sub-second tolerance levels, $\delta \in \{0.1, 0.2, 0.3\}$ s. Against a median cycle period of roughly 0.4 s, AP@0.1 is the strict regime, in which a correct match must land within a half-cycle window of its referent; AP@0.2 and AP@0.3 are looser and admit some drift toward neighboring cycles. We report the mean across all three so the metric grades both precise cycle attribution and coarser localization, with AP@0.1 as the strictest of the three. The mAP aggregation jointly rewards completeness across the full cycle sequence and precision in placing each prediction near the correct cycle, and penalizes degenerate strategies such as enumerating evenly spaced timestamps to cover the clip.

[1] https://huggingface.co/datasets/lmms-lab-encoder/JumpScore

Table 1 Video understanding benchmarks for 8B-class MLLMs. LLaVA-OneVision-2-8B leads the 18-task video understanding average (62.5) and the 4-task RVOS tracking average (48.0, J&F). We use and to denote the best and second-best results.

Group	Benchmark	LLaVA-OV-2	Qwen3-VL	Keye-VL-1.5	InternVL-3.5	PLM	LLaVA-OV-1.5
Video QA	MV-Bench	66.2	69.0	56.9	72.1	**77.1**	51.2
	NextQA	82.5	83.4	75.8	82.0	**84.1**	73.7
	TempCompass-MC	74.5	74.3	**75.5**	70.4	72.7	57.5
	VideoMME	71.9	71.4	**73.0**	65.9	60.5	61.1
	VideoMME (w/ sub.)	**76.3**	75.6	76.2	68.6	65.6	65.5
	VideoMME-v2-64	**19.9**	18.2	14.1	14.6	8.7	9.1
	MLVU-dev	76.6	**78.1**	75.0	71.0	66.4	62.1
	LVBench	55.5	**58.0**	42.8	46.7	44.5	40.1
	LongVideoBench	66.9	**68.0**	66.0	62.4	59.6	56.2
	MMVU-val	56.2	58.7	**68.3**	60.2	43.3	50.1
	VideoEval-Pro	**61.5**	59.2	54.9	50.1	47.2	44.8
	MMOU	39.5	**40.6**	35.3	36.1	26.2	30.7
	JumpScore	**74.9**	30.1	39.6	11.0	13.1	2.1
Temp. Ground.	t/Charades	**53.5**	48.3	45.4	27.8	34.5	15.6
	t/ActivityNet	**53.8**	46.8	41.3	31.3	7.6	17.7
	t/QVHighlights	**66.4**	59.4	55.5	31.3	4.2	21.0
Vis. Spatial	VSI-Bench	**70.9**	59.1	36.4	56.0	27.9	30.2
	ReVSI	**57.6**	48.9	32.4	47.9	30.7	33.5
Video Average (18 tasks)		**62.5**	58.2	53.6	50.3	43.0	40.1
Tracking (J&F)	DAVIS	**58.7**	41.3	5.8	4.7	2.0	4.1
	MeViS_U	**45.7**	28.4	7.2	7.5	7.6	6.1
	ReVOS-ref	**58.2**	37.8	10.7	10.2	8.5	13.0
	ReVOS-reason	**29.2**	21.9	9.6	9.2	10.2	9.7
Tracking Average (4 tasks, J&F)		**48.0**	32.4	8.3	7.9	7.1	8.2

7 Evaluation

This section reports end-to-end results for LLaVA-OneVision-2 against 8B-class open MLLMs across three benchmark domains targeted by our recipe: *video understanding*, *spatial reasoning*, and *image and document understanding* (§7.1). We further evaluate referring video object segmentation as a robotics-relevant test of temporally coherent spatial grounding and object-identity consistency. The codec-versus-uniform-sampling analysis underlying the video gains is reported separately in §8.

7.1 Main Multimodal Benchmark Results

We evaluate **LLaVA-OV-2-8B** on a comprehensive suite of public benchmarks spanning *video understanding*, *spatial reasoning*, and *image and document understanding*. We compare against four 8B-class baselines: Qwen3-VL-8B, Keye-VL-1.5-8B, InternVL-3.5-8B, and LLaVA-OV-1.5-8B; PLM-8B is included where results are available. Unless otherwise specified, all numbers are obtained with LMMs-Eval (Zhang et al., 2025c) using default prompts, identical decoding strategies, and matched token budgets across compared models.

Video Understanding. Table 1 reports results on 18 video benchmarks covering short-clip QA (MV-Bench, NextQA, TempCompass), full-length VideoMME (with and without subtitles, plus VideoMME-v2 (Fu et al.,

Table 2 Spatial reasoning, image and document benchmarks for 8B-class MLLMs. LLaVA-OneVision-2-8B is most differentiated on spatial reasoning (notably **CrossPoint**, **TraceSpatial-3D**), while staying competitive on image / document understanding and leading V*-Bench. We use and to denote the best and second-best results.

Group	Benchmark	LLaVA-OV-2	Qwen3-VL	Keye-VL-1.5	InternVL-3.5	PLM	LLaVA-OV-1.5
Spatial Reasoning	CRPE	77.3	**77.7**	75.2	75.0	77.0	74.8
	MetaVQA	**69.1**	68.7	59.2	65.7	45.4	67.1
	ERQA	43.3	42.3	38.3	41.8	**44.3**	41.5
	CV-Bench 2D	**82.6**	81.0	78.2	77.9	80.6	76.5
	CV-Bench 3D	**92.8**	92.3	82.0	86.3	82.4	82.9
	CrossPoint	**61.9**	26.9	20.2	20.2	15.7	15.9
	EmbSpatial	**78.1**	77.5	66.3	73.2	73.5	64.2
	SAT	**69.3**	**69.3**	62.7	54.7	36.7	61.3
	MMSI-Bench	29.6	31.0	26.7	28.1	**31.4**	28.3
	BLINK	63.5	**65.1**	52.2	55.7	56.0	48.3
	TraceSpatial-3D	**31.0**	8.0	3.0	4.0	1.0	1.0
Spatial Average (11 tasks)		**63.5**	58.2	51.3	53.0	49.5	51.1
Image & Document	MMStar	64.8	62.9	**73.6**	66.6	57.9	67.9
	MMBench-en	85.7	84.9	**88.5**	87.9	80.2	85.6
	DocVQA	95.2	95.7	94.9	92.3	94.6	**97.8**
	ChartQA	85.9	85.1	84.7	**86.7**	85.5	86.5
	InfoVQA	74.4	**83.4**	76.9	79.1	80.0	79.1
	OCRBench	78.2	84.7	**84.8**	84.0	83.2	82.6
	AI2D	84.3	83.6	86.0	84.0	**92.7**	84.0
	V*-Bench	**85.9**	85.3	78.0	81.7	71.2	77.5
	CountBench	89.0	89.8	83.1	75.6	**91.8**	87.8
	Pixmo-Count	64.0	62.4	55.6	61.8	**68.0**	63.1
	RealWorldQA	69.7	69.4	69.8	63.1	**72.7**	68.1
Image & Document Average (11 tasks)		79.7	**80.7**	79.6	78.4	79.8	80.0

2026)), long-video understanding (MLVU-dev, LVBench, LongVideoBench), expert and omni-modal reasoning (MMVU-val, VideoEval-Pro, MMOU), temporal grounding (Charades-STA, ActivityNet, QVHighlights), visual–spatial intelligence in video (VSI-Bench (Yang et al., 2024), ReVSI (Zhang et al., 2026d)), and **JumpScore**, for which we report JumpScore mAP. JumpScore targets cycle-level temporal localization in high-frequency repetitive motion, where models must distinguish the correct action instance among many visually similar motion cycles. LLaVA-OneVision-2-8B achieves the highest average score across this 18-task suite, improving over Qwen3-VL-8B by +4.3 points (62.5 vs. 58.2). The largest gains appear on the axes targeted by our recipe: temporal grounding (Charades-STA: +5.2, ActivityNet: +7.0, QVHighlights: +7.0), visual–spatial video reasoning (VSI-Bench: +11.8, ReVSI: +8.7), and JumpScore (74.9 vs. 30.1 JumpScore mAP, +44.8 over Qwen3-VL-8B). These results indicate that codec-stream tokenization and shared 3D RoPE improve event-level video evidence allocation under irregular spatiotemporal token layouts.

LLaVA-OneVision-2-8B leads the 8B class on the 18-task video suite, with the largest gains on temporal grounding, visual–spatial video reasoning, and JumpScore temporal localization.

Spatial Reasoning. We next evaluate spatial reasoning, the axis most directly targeted by the Stage-4 spatial recipe. Table 2 reports 11 spatial benchmarks spanning relation comprehension, embodied scene understanding, 2D/3D spatial reasoning, trajectory reasoning, cross-view correspondence, spatial aptitude, multi-image spatial

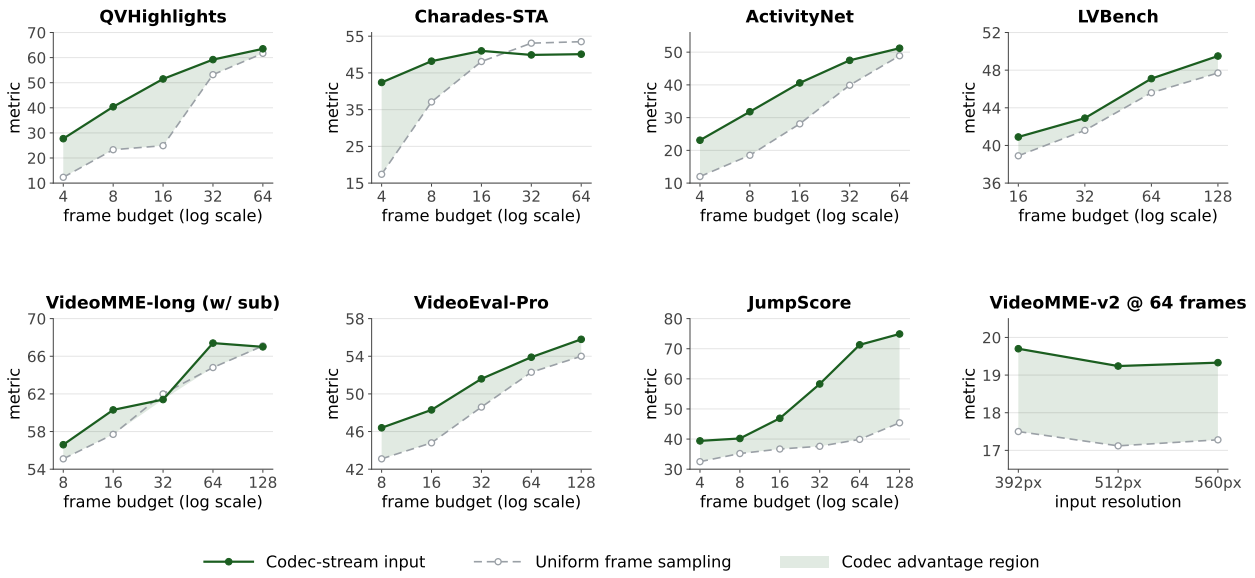


Figure 7 Codec-stream inputs versus frame sampling. Under matched visual settings, codec-stream tokenization improves event-level temporal grounding by following high-bit-cost intervals and high-residual regions rather than uniformly sampled frame slots. We evaluate this effect on temporal grounding benchmarks, long-form video QA, and JumpScore, our fine-grained temporal-localization benchmark for high-frequency repeated motion.

intelligence, and general perception. LLaVA-OneVision-2-8B achieves the best spatial average, improving over Qwen3-VL-8B by +5.3 points on this 11-task subset (63.5 vs. 58.2). Its largest gains occur where existing 8B baselines are weakest: CrossPoint improves by +35.0 points over Qwen3-VL-8B (61.9 vs. 26.9), and TraceSpatial-3D reaches nearly 4× the next-best 8B score (31.0 vs. 8.0). The model further leads or matches the 8B class on MetaVQA, CV-Bench 2D/3D, EmbSpatial, and SAT, while remaining competitive on CRPE, ERQA, MMSI-Bench, and BLINK. We attribute these gains to the Stage-4 spatial corpus and point-based tracking supervision, which explicitly train structured spatial outputs beyond caption-style perception.

> Stage-4 spatial supervision yields the largest gains where 8B baselines are weakest: +35.0 on CrossPoint and nearly 4× the next-best 8B score on TraceSpatial-3D.

Image and Document Understanding. We also evaluate whether long-video and spatial training preserve image and document capability. Table 2 summarises results on 11 image and document benchmarks. LLaVA-OneVision-2-8B remains competitive on DocVQA (95.2), ChartQA (85.9), CountBench (89.0), Pixmo-Count (64.0), and RealWorldQA (69.7), and leads the 8B class on V*-Bench (85.9). On text-dense or diagram-heavy tasks, it trails the strongest specialized baselines: Keye-VL-1.5 leads MMStar and OCRBench, Qwen3-VL-8B leads InfoVQA, and PLM-8B leads AI2D. These gaps indicate that the spatially grounded long-video recipe preserves strong image-level capability, but does not make the model OCR- or document-specialized.

> LLaVA-OneVision-2 preserves strong image and document capability, leading V*-Bench while remaining below OCR- and document-specialized baselines on text-dense tasks.

8 Ablation of Codec-Stream Tokenization

Codec-stream tokenization is designed to make video observation follow the predictive structure of the compressed stream. To isolate this effect, we compare codec-stream inputs with frame-sampling inputs while keeping the backbone, language model, decoder, prompts, and evaluation protocol fixed. The results show that codec-stream tokenization is most effective when the answer depends on event-level evidence: Codec-stream yields a 17.3-point average gain on JumpScore and a 9.7-point average gain across three temporal-grounding

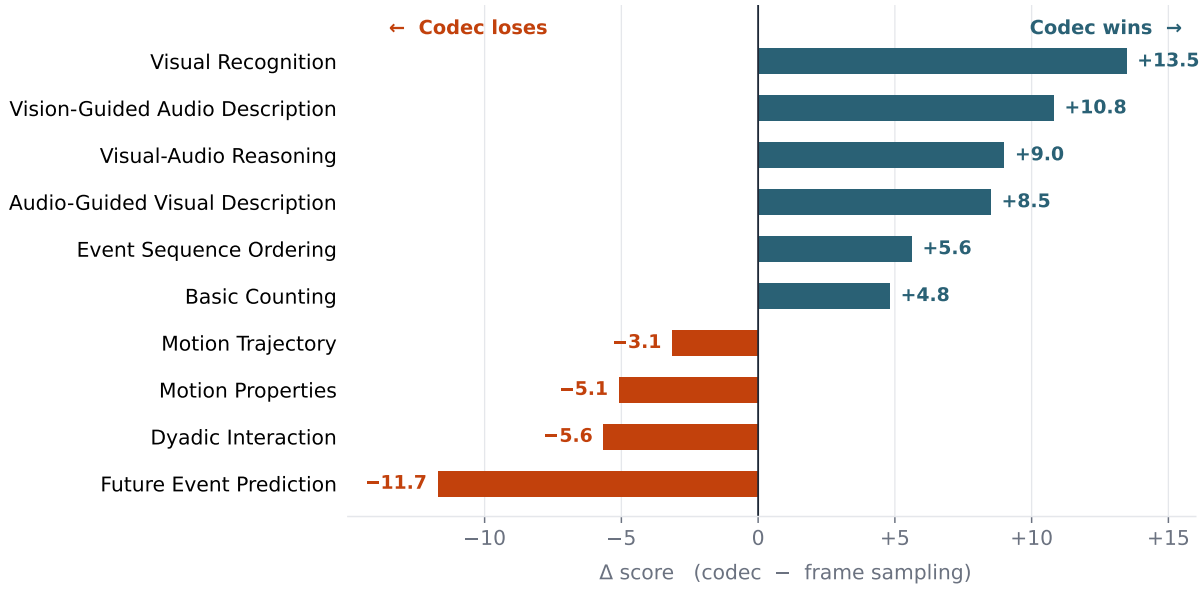


Figure 8 Per-skill ablation on VideoMME-v2 at matched visual settings. Teal bars indicate capabilities where codec-stream inputs outperform frame sampling; coral bars indicate capabilities where frame sampling is stronger.

benchmarks over uniform frame sampling. On long-form video QA, codec-stream inputs preserve parity or small gains over frame sampling, indicating that stream-aware evidence allocation does not trade off long-video semantic understanding for compression.

Temporal Grounding. Temporal grounding exposes the core advantage of codec-stream tokenization. Uniform frame sampling allocates observation capacity by elapsed time, so short event intervals can be missed when they fall between sampled frames. Codec-stream tokenization instead uses bit-cost dynamics to identify temporally informative segments and motion-residual cues to retain spatial evidence around perceptual transitions. As shown in Figure 7, this produces the largest gains at low input scales, where uniform sampling has the highest probability of missing event boundaries. The gap narrows as the input scale increases, since denser frame sampling gradually recovers more event evidence. Averaged over all temporal-grounding settings, codec-stream improves the score from 35.5 to 45.2, yielding a 9.7-point average absolute gain over uniform frame sampling.

> Codec-stream tokenization improves temporal grounding by biasing visual evidence toward high-bit-cost transitions and high-residual regions, rather than distributing tokens uniformly over time.

Fine-grained Grounding in Densely Repeated Motion. JumpScore targets a complementary failure mode of existing long-video benchmarks: high-frequency, densely repeated motion, where the challenge is not recognizing the event category but localizing the correct action instance among many visually similar cycles. In this regime, periodic frames can appear redundant under visual similarity, even though their cycle boundaries carry the decisive temporal signal. Codec-stream tokenization is well matched to this setting because bit-cost spikes and residual responses concentrate around cycle transitions. On JumpScore, LLaVA-OneVision-2-8B reaches 74.9 JumpScore mAP, outperforming Qwen3-VL-8B (30.1) by +44.8 points; under matched visual budgets on the same benchmark, codec-stream inputs improve temporal grounding over frame sampling by +17.3 points.

> On densely repeated motion, codec-stream tokenization localizes perceptual transitions between visually similar cycles, the regime where uniform frame sampling and similarity-based deduplication are least reliable.

Long-form Video Question Answering. Long-form video QA is a more forgiving setting for frame sampling,

Table 3 Extended long-video frame-budget sweep on LVBench, VideoMME-Long, MLVU-dev, and VideoEval-Pro. Stream inputs hold parity or a small lead at low budgets and converge closely with Fix inputs at higher budgets.

Token Budget	LVBench		VideoMME-L (w/ sub)		MLVU-dev		VideoEval-Pro	
	Fix	Stream	Fix	Stream	Fix	Stream	Fix	Stream
2k	**40.0**	38.5	55.1	**56.6**	60.0	**62.4**	45.0	**46.4**
4k	40.4	**40.9**	57.7	**60.3**	62.6	**65.0**	45.3	**48.3**
8k	**43.4**	42.9	**62.0**	61.4	67.1	**69.0**	48.1	**51.6**
16k	46.7	**47.1**	64.8	**67.4**	70.2	**71.3**	51.4	**53.9**
32k	48.8	**49.5**	**67.1**	67.0	72.5	**73.7**	**56.0**	55.8

since many answers can be recovered from sparse semantic snapshots. The relevant question is therefore whether codec-stream tokenization introduces a regression by prioritizing motion-residual evidence. Figure 7 shows that it does not: codec-stream inputs maintain parity or small gains on LVBench, VideoMME-Long with subtitles, and VideoEval-Pro. This suggests that codec-stream tokenization preserves broad semantic coverage while reallocating additional capacity toward event-bearing moments. At higher input scales, the two strategies converge, indicating that abundant frame evidence can partially compensate for less adaptive allocation.

> On long video QA, codec-stream tokenization preserves the semantic robustness of frame sampling while allocating additional evidence to stream-indicated event moments.

Figure 8 further clarifies the competence boundary of codec-stream tokenization. Codec-stream inputs gain on tasks whose decisive evidence is a salient spatial snapshot or a discrete state transition, including visual recognition, event ordering, counting, and several audio-visual reasoning skills. Frame sampling remains stronger when the task requires dense trajectory continuity, such as future event prediction, dyadic interaction, motion properties, and motion trajectory estimation. This pattern supports the coverage–resolution trade-off identified in our analysis: codec-stream inputs expand event-level coverage, while frame sampling preserves frame-level evidential resolution for detail-sensitive motion reasoning.

Fixed Versus Stream-adaptive Codec Allocation. To isolate the contribution of stream adaptivity inside the codec pipeline, we compare Fix, which follows a fixed GOP/slot schedule, with Stream, which derives temporal groups and key-frame anchors from the continuous bit-cost stream and packs motion-residual evidence into compact visual canvases. Across LVBench, VideoMME-Long, MLVU-dev, and VideoEval-Pro, Stream matches or improves over Fix in most settings, with especially consistent gains on MLVU-dev and VideoEval-Pro. The result supports our central design claim: the advantage of codec-stream tokenization comes not merely from compression, but from making visual observation follow bit-cost dynamics and motion-residual structure rather than a preset GOP layout.

9 Related Work

LLaVA-OneVision-2 sits at the intersection of five lines of work: open video MLLMs (§9.1), patch- and token-level efficiency for video transformers (§9.2), temporal grounding with video LLMs (§9.3), spatial cognition in multimodal LLMs (§9.4), and referring video object segmentation (§9.5). We review each in turn and position codec-stream tokenization within them.

9.1 Open Video MLLMs

The 8B-class open multimodal model has converged into a recognisable design template: a vision transformer backbone (typically SigLIP-style or a derivative), a connector projecting visual patches into the language-model embedding space, and an instruction-tuned language model (Qwen, LLaMA, InternLM). Recent representative releases at this scale include Qwen2-VL / Qwen2.5-VL / Qwen3-VL (Wang et al., 2024b; Bai et al., 2025b,a),

InternVL-3 / InternVL-3.5 (Zhu et al., 2025; Wang et al., 2025b), Keye-VL-1.5 (Yang et al., 2025a), NVILA (Liu et al., 2024b), VideoLLaMA 3 (Zhang et al., 2025a), MiniCPM-V (Yao et al., 2024), Aria (Li et al., 2024), Eagle 2 (Li et al., 2025d), Apollo (Zohar et al., 2024), Oryx-MLLM (Liu et al., 2024a), Tarsier 2 (Yuan et al., 2025b), LongVU (Shen et al., 2024), LongVILA (Chen et al., 2024b), InternVideo2.5 (Wang et al., 2025c), Penguin-VL (Zhang et al., 2026a), and the LLaVA-OneVision-1.5 release (Li et al., 2025a; An et al., 2025) that LLaVA-OneVision-2 inherits from directly. Most of these models share the same default observation strategy on video input: a clip is reduced to 8–32 uniformly-spaced frames, every patch in each sampled frame is encoded, and the rest of the timeline is discarded. The Cambrian-1 (Tong et al., 2024) and Cambrian-S (Yang et al., 2025c) releases additionally probe the failure modes of this default, particularly long-horizon continual reasoning, without altering the underlying frame-sampling prior. VideoChat (Li et al., 2023; Maaz et al., 2024), ShareGPT4Video (Chen et al., 2024a), and the LLaVA-Video instruction-data line (Zhang et al., 2024; Maaz et al., 2024) are likewise built on top of uniform sampling. Our contribution is orthogonal to this template: codec-stream tokenization is a substitution at the input side of the visual encoder, leaving the connector, language model, and instruction-tuning recipe unchanged. Any of the systems above can in principle adopt it without touching their downstream stack.

9.2 Efficient Video Tokenization

A large body of prior work has tackled the same redundancy problem we identify, that most patches across most frames are not informative, but from a different direction. Three families of approaches deserve comparison.

Token dropout and merging. Token dropout (Rao et al., 2021; Yin et al., 2022; Meng et al., 2022) removes a learned subset of tokens at intermediate transformer layers; token merging (Bolya et al., 2022; Bolya and Hoffman, 2023) combines similar tokens via attention-key similarity. A second family of recent work pushes this idea into the video-MLLM stack itself: hierarchical clip-to-video compression in VideoChat-Flash (Li et al., 2025b), one-token-per-frame summarisation in LLaVA-Mini (Zhang et al., 2025d), Mamba-based temporal pooling in STORM (Jiang et al., 2025), dual-stream slow-fast token routing (Xu et al., 2025a), training-free redundancy pruning in ReTaKe (Wang et al., 2024d), and reconstructive or KV-sparsification compression in the Video-XL family (Shu et al., 2025a; Liu et al., 2025; Shu et al., 2025b). Concurrent 2026 entries continue this trajectory: visual KV-cache memory mechanisms (FlexMem (Chen et al., 2026c)), small-VLM-as-local-compressor pipelines (Tempo (Fei et al., 2026)), extreme one-token-per-frame compression (Zhang et al., 2026e), density-adaptive samplers (Chen et al., 2026d), RL-learned per-token retention (SCORE (Wang et al., 2026)), semantic-guided evolutionary compression (EvoComp (Song et al., 2026)), and hybrid KV-cache compression (Zeng et al., 2026). All of these families typically operate at the model side (inside the transformer stack or the LLM-side KV cache) rather than at the input side, and require either learned routing, a similarity-based pooling step, or a task-conditional sparsification policy that adds inference-time computation. Codec-stream tokenization differs in two ways: it operates entirely at the input side, so any downstream model can be a plain transformer with no routing logic; and it derives the saliency signal from the codec bitstream rather than from learned model activations or post-hoc similarity, so the selection is computed once at preprocessing time and is agnostic to the downstream model.

Compressed-domain video tokenization. A closer line of work directly exploits the codec representation. VideoLaVIT (Jin et al., 2024) decomposes videos into keyframes and motion vectors for unified generative pretraining. Run-Length Tokenization (RLT) (Choudhury et al., 2025) exploits temporal redundancy at the token level. EMA (Zhao et al., 2025) encodes GOP structures with motion-aware mechanisms for efficient video MLLM understanding. AuroraLong (Xu et al., 2025b) brings RNN-style token compression back for efficient long-form video understanding. Concurrent 2026 work has further explored this direction: ReMoRa (Yashima et al., 2026) operates on sparse RGB I-frames augmented with refined motion-vector representations through a hierarchical motion state-space module, and a recent survey (Jin et al., 2026b) argues for the joint standardisation of traditional visual coding (H.264/H.265) and visual token technology. Each of these methods extracts a different signal from the compressed bitstream and demonstrates a different efficiency–accuracy trade-off; what is missing in all of them is a unified *operational* pipeline that runs at training and inference under the same configuration, makes the signal-extraction choices explicit and tunable (variable-length GOP, bit-cost vs. MV+residual scoring, stratified block selection), and Pareto-dominates uniform sampling across temporal grounding, R-VOS, and 3D-RoPE token-efficiency under matched token budgets. §2.2 provides exactly that

pipeline; §8 provides the controlled empirical comparison across temporal grounding, counting, long-form QA, token-efficiency under 3D-RoPE, and per-skill diagnostics on VideoMME-v2.

Adaptive frame sampling. A simpler line of prior work uses learned or heuristic adaptive frame samplers (Tang et al., 2025; Wang et al., 2025a) to allocate more frames to motion-intensive segments. The canonical limitation is that the unit of selection is the frame, not the patch: a single frame contributes either every patch or no patches, and the quadratic cost of self-attention is unmitigated for the kept frames. Codec-stream tokenization instead selects at the patch level under a *global* clip-level token budget, so a high-motion frame can contribute a small dense patch cluster and a low-motion frame can contribute its I-frame anchor only.

9.3 Temporal Grounding with Video LLMs

A complementary line of work investigates how video LLMs can localise events in time rather than only describing them. Early approaches augment a frame-sampled video LLM with explicit timestamp tokens or boundary-aware training, including VTimeLLM (Huang et al., 2024a), TimeChat (Ren et al., 2023), LITA (Huang et al., 2024b), VTG-LLM (Guo et al., 2024b), and Grounded-VideoLLM (Wang et al., 2024a), which add per-event captioning supervision or specialised time-token vocabularies on top of the standard uniform-frame interface. TRACE (Guo et al., 2024a) and TimeRefine (Yan et al., 2024) push this further with causal event modelling and iterative boundary refinement. More recent work casts temporal grounding as a reinforcement-fine-tuning problem with verifiable IoU-based rewards: VideoChat-R1 (Li et al., 2025c), Time-R1 (Wang et al., 2025d), and Video-R1 (Feng et al., 2025) all post-train a frame-sampled MLLM with GRPO-style updates against grounding rewards. The 2026 cohort extends this trajectory with instance-level grounding (STVG-R1 (Zhang et al., 2026c)), curriculum grounding rewards (Video-TwG (Chen et al., 2026a)), event-graph reasoning rewards (GraphThinker (Cheng et al., 2026)), and verifiable-reward video reasoning on flow-based generative models (Wan-R1 (Liu et al., 2026)), alongside broader unified RL post-training recipes for vision-language models (Yan et al., 2026; Xie et al., 2026). All of these systems retain uniform frame sampling on the input side; their gains come from the supervision objective, not the visual evidence allocation. Codec-stream tokenization is orthogonal: it changes which frames and which patches the model sees in the first place. The two are stackable, and on JumpScore, Charades-STA, ActivityNet, and QVHighlights (§8) we show that codec-stream inputs alone, with no temporal-grounding-specific objective, already match or exceed frame-sampled grounding baselines under matched token budgets.

9.4 Spatial Cognition in Multimodal LLMs

Spatial cognition has emerged as a distinct evaluation axis where 8B-class video MLLMs underperform their headline results. Cambrian-S (Yang et al., 2025c) formalises "supersensing" — maintaining coherent spatial state across long-horizon video — as the next bottleneck for video MLLMs and identifies inference-time mechanisms (predictive sensing with surprise-driven memory management) as the most promising direction; we read LLaVA-OneVision-2 as a strong spatially-grounded *base* for that line of work rather than a competing solution to it. Molmo and Pixmo (Deitke et al., 2024) introduced point-based supervision as a structured-output format for static images; Molmo2 (Allen Institute for AI, 2025; Clark et al., 2026) extends this to point-based video tracking and spatio-temporal pointing. We adopt the Molmo2 video-tracking and pointing data *as-is* as one ingredient of our Stage-4 spatial supervision (§4.4); the contribution this paper isolates is codec-stream tokenization, not the point/tracking signal itself. A parallel data-side line of work supplies structured 2D and 3D spatial supervision for MLLMs: VSI-Bench and the "Thinking in Space" analysis (Yang et al., 2024), SpatialBot (Cai et al., 2024), SpatialRGPT (Cheng et al., 2024), EmbodiedScan (Wang et al., 2024c), RoboSpatial (Song et al., 2024), RoboRefer / RefSpatial (Zhou et al., 2025), SPAR (Zhang et al., 2025b), MMSI-Bench (Yang et al., 2025b), and the SG-RLVR recipe of SpaceR (Ouyang et al., 2025). Concurrent 2026 work continues this thread: GR3D (Yuan et al., 2026) attaches geometrically referenced 3D scene representations to MLLMs without training, Spa3R (Jiang et al., 2026) learns view-invariant spatial fields via predictive novel-view feature synthesis, and "Thinking with Spatial Code" (Chen et al., 2026b) feeds explicit oriented-bounding-box code into the LLM for physical-world video reasoning. New benchmarks tighten the evaluation protocol: ReVSI (Zhang et al., 2026d) re-annotates VSI-Bench scenes to remove protocol artefacts (we report ReVSI in §7.1), VAEX-Bench (Bang and Song, 2026) probes abstractive spatiotemporal evidence at object/room/floor-plan level, and SFI-Bench (Zhang et al., 2026b) jointly benchmarks spatial and

functional intelligence over video. These datasets converge on a similar three-source taxonomy (annotated real videos, simulated trajectories, pseudo-annotated web video) that our `LLaVA-OneVision-2-Spatial-4M` corpus (§4.4) follows. The novelty on the data side is not the taxonomy but the decision to pair structured spatial QA with Molmo2-style point/tracking supervision exclusively in Stage 4 of the recipe; §4.4 explains why this stage ordering matters empirically.

9.5 Referring Video Object Segmentation with Multimodal LLMs

Referring video object segmentation (R-VOS) on MeViS (Ding et al., 2023) and ReVOS asks the model to follow a single physical reference across many frames given a language query, and is the closest public proxy for the cross-frame coherence that robotics deployment needs. Recent MLLM-based R-VOS systems pair an LLM with a promptable segmentation backbone, typically SAM 2 (Ravi et al., 2024): VideoLISA (Bai et al., 2024) introduces a single `<TRK>` token under a sparse-dense sampling regime, Sa2VA (Yuan et al., 2025a) marries SAM 2 with LLaVA-OneVision for unified image / video grounding, VideoGLaMM (Munasinghe et al., 2025) extends GLaMM (Rasheed et al., 2024) to pixel-level video grounding, Vitron (Fei et al., 2024) unifies twelve image/video tasks under one pixel-level LLM, and VideoMolmo (Rasheed et al., 2025) couples Molmo-style point supervision with bidirectional SAM 2 propagation. Concurrent 2026 work pushes toward agentic and tracking-aware R-VOS: AgentRVOS (Jin et al., 2026a) first generates SAM 3 object tracks and then asks the LLM to reason over them for zero-shot R-VOS, while SPARROW (Alansari et al., 2026) learns dual box/segmentation prompts with tracked features for spatially precise, temporally consistent pixel grounding.

10 Conclusion

We presented **LLaVA-OneVision-2** as a codec-aligned long-video MLLM that moves video observation beyond frame-centric sampling toward stream-aware perceptual evidence allocation. Built on a native-resolution OneVision-Encoder, the model treats compressed video as a continuous bit-cost stream: bit-cost dynamics determine adaptive temporal groups, while motion-residual evidence is condensed into compact visual canvases and processed through a unified group-visible attention interface. Together with a progressive open training stack of approximately 8M re-captioned video samples, a 4M-sample 2D&3D spatial corpus, and the JumpScore temporal-localization benchmark, LLaVA-OneVision-2 demonstrates strong gains across video, spatial, temporal grounding, and tracking evaluations. More importantly, our analysis shows that codec-stream tokenization is not merely a token-reduction mechanism, but a perceptual allocation principle: it expands event-level coverage for long-video reasoning while retaining frame sampling as a complementary path for detail-sensitive perception. Looking forward, we will extend this codec-aligned paradigm toward streaming perception and hours-scale or longer codec-context modeling (Guan et al., 2026; Shen et al., 2026; Chen et al., 2026c), where visual evidence must be continuously updated, compressed, and retrieved over ultra-long temporal horizons.

11 Contributors

Contributors

core contributors are in bold

- **Xiang An**
- **Yin Xie**
- **Feilong Tang**
- **Yunyao Yan**
- **Huajie Tan**
- **Didi Zhu**
- **Changrui Chen**
- **Xiuwei Zhao**
- Bin Qin
- Kaicheng Yang
- Yifei Shen
- Yuanhan Zhang
- Kaichen Zhang
- Wenkang Zhang
- Zheng Cheng
- Chunsheng Wu
- Chunjiang Ge
- Zimin Ran
- Dehua Song
- Chunyuan Li
- Shikun Feng
- Ming Hu
- Zhangquan Chen
- Junbo Niu

Project Leaders

- Bo Li
- Ziyong Feng
- Ziwei Liu
- Zongyuan Ge
- Jiankang Deng

12 Details

12.1 Codec-Stream versus Uniform Frame Sampling

Figure 7 compares codec-stream inputs with uniformly sampled RGB frames under matched nominal frame budgets. To make the scaling curves auditable, we report the exact numerical values used in the figure in Tables 4 and 5. "Uniform" denotes standard uniform frame sampling, and "Codec" denotes the codec-stream input representation.

Table 4 Numerical values for the video QA and JumpScore curves in Figure 7. Each benchmark is reported under matched nominal frame budgets.

Budget	VideoMME-L-sub		LVBench		VideoEval-Pro		JumpScore	
	Uniform	Codec	Uniform	Codec	Uniform	Codec	Uniform	Codec
4	–	–	–	–	–	–	32.5	39.4
8	55.1	56.6	–	–	43.1	46.4	35.2	40.2
16	57.7	60.3	38.9	40.9	44.8	48.3	36.7	46.9
32	62.0	61.4	41.6	42.9	48.6	51.6	37.6	58.3
64	64.8	67.4	45.6	47.1	52.3	53.9	39.9	71.3
128	67.1	67.0	47.7	49.5	54.0	55.8	45.4	74.9

Table 5 Numerical values for the temporal grounding curves in Figure 7. Each benchmark is reported under matched nominal frame budgets.

Budget	QVHighlights		Charades-STA		ActivityNet Captions	
	Uniform	Codec	Uniform	Codec	Uniform	Codec
4	12.3	27.7	17.4	42.4	12.0	23.1
8	23.3	40.4	37.1	48.2	18.5	31.8
16	24.9	51.5	48.1	51.0	28.1	40.6
32	53.2	59.2	53.1	49.9	39.9	47.5
64	61.7	63.5	53.5	50.1	48.9	51.2

Across the general long-video QA benchmarks, codec-stream inputs provide modest but mostly positive gains over uniform frame sampling. The average absolute improvements are 1.2 points on VideoMME-L-sub, 1.7 points on LVBench, and 2.6 points on VideoEval-Pro. The gains are not strictly monotonic at every budget: for example, VideoMME-L-sub shows small drops at 32 and 128 frames. Nevertheless, the overall pattern suggests that codec-stream tokenization can preserve useful temporal evidence without increasing the nominal frame budget.

The advantage becomes larger on motion-sensitive temporal localization tasks. On JumpScore, codec-stream improves the average score from 37.9 to 55.2 across the six evaluated budgets, corresponding to an average absolute gain of 17.3 points. At the 4-frame budget, the score improves from 32.5 to 39.4, which corresponds to 6.9 absolute points and 21.2% relative improvement over uniform frame sampling. The gains further increase at larger budgets, reaching 20.7, 31.4, and 29.5 points at 32, 64, and 128 frames, respectively.

For temporal grounding, codec-stream is most beneficial under low- and medium-frame budgets. On QVHighlights, the gains are 15.4, 17.1, and 26.6 points at 4, 8, and 16 frames, respectively. On Charades-STA, codec-stream substantially improves the 4-frame setting from 17.4 to 42.4, while uniform sampling becomes competitive at higher budgets. On ActivityNet Captions, codec-stream improves over uniform sampling at all evaluated budgets, with gains ranging from 2.3 to 13.3 points. Overall, the numerical values behind Figure 7 show that codec-stream inputs provide the largest benefit when temporally sparse, motion-relevant evidence must be recovered from a limited visual budget.

12.2 Video Tracking

Tracking protocol. Across all benchmarks, segmentation metrics are computed at the original video frame rate, whereas point-based metrics are evaluated at 1 fps and marked as correct when a predicted point falls inside the ground-truth mask. For segmentation-capable baselines, we evaluate the masks produced by the corresponding open segmentation models and report their mask-level quality. When a model can produce discrete points for the referred object, such as VLM-style point outputs, we additionally evaluate its point-based localization accuracy. LLaVA-OneVision-2-8B does not introduce a new segmentation head. Instead, it predicts discrete point tracks with explicit object IDs, and these ID-associated points are directly used as SAM 2 point prompts to obtain per-frame segmentation masks. The reported mask metrics for our model therefore evaluate the quality of ID-consistent point tracking together with the point-to-mask conversion, rather than a separately trained segmentation decoder.

Table 6 Referring video object segmentation and point-to-mask tracking. Evaluation on DAVIS, MeViS-U, REVOS-Referring, and REVOS-Reasoning with three metrics: F, HOTA, and J&F. For LLaVA-OneVision-2-8B, predicted ID-associated point tracks are converted into masks using SAM 2 point prompts before computing mask-level metrics. The final Overall column reports the aggregate tracking score used by our evaluation protocol, rather than the unweighted mean of the four J&F columns. **Bold** indicates the best result in each column.

	DAVIS			MeViS-U			REVOS (Ref.)			REVOS (Reason)			Overall
Method	F	HOTA	J&F	F	HOTA	J&F	F	HOTA	J&F	F	HOTA	J&F	
InternVL3.5-8B	12.8	9.4	4.7	7.2	6.6	7.5	22.2	20.8	10.2	7.9	7.2	9.2	11.0
LLaVA-OV-1.5	11.9	8.8	4.1	7.3	9.0	6.1	16.8	17.4	13.0	6.2	17.8	9.7	10.8
Keye-VL-1.5	14.6	12.2	5.8	10.1	15.8	7.2	22.1	26.5	10.7	9.9	12.0	9.6	12.1
PLM-8B	7.8	32.8	2.0	5.0	34.4	7.6	6.8	51.2	8.5	0.1	0.1	10.2	9.0
Qwen3-VL-8B	39.7	36.6	41.3	29.9	**34.5**	28.4	40.7	41.5	37.8	24.7	**32.4**	21.9	30.8
LLaVA-OV-2-8B	**52.7**	**44.7**	**58.7**	**37.1**	31.7	**45.7**	**60.8**	**56.1**	**58.2**	**27.4**	22.5	**29.2**	**41.0**

Results. Table 6 shows that LLaVA-OneVision-2-8B achieves the best overall tracking score, improving over Qwen3-VL-8B by **+10.2** absolute points (41.0 vs. 30.8). The gain is consistent on the mask-overlap metric J&F across all four evaluation splits: +17.4 on DAVIS (58.7 vs. 41.3), +17.3 on MeViS-U (45.7 vs. 28.4), +20.4 on REVOS-Referring (58.2 vs. 37.8), and +7.3 on REVOS-Reasoning (29.2 vs. 21.9). These results support our hypothesis that codec-stream dense observation is useful for cross-frame correspondence: at matched token budgets, retaining I-frame patches densely while sparsifying P-frames toward motion-relevant regions provides more temporally aligned visual evidence per token. The Stage-4 point-based tracking supervision from Molmo2-VideoTrack provides a complementary mechanism, teaching the model to maintain a single physical reference across frames instead of re-grounding the object from text independently at every frame. Although Qwen3-VL-8B remains stronger on HOTA for MeViS-U and REVOS-Reasoning, LLaVA-OneVision-2-8B obtains higher J&F on every split, indicating better mask overlap after point-to-mask conversion.

13 Case Study

13.1 Temporal Grounding on TimeLens-Bench

Figure 9 reports ten temporal-grounding cases from Charades-STA, ActivityNet-Captions, and QVHighlights; per-row metric is mean IoU over five runs.

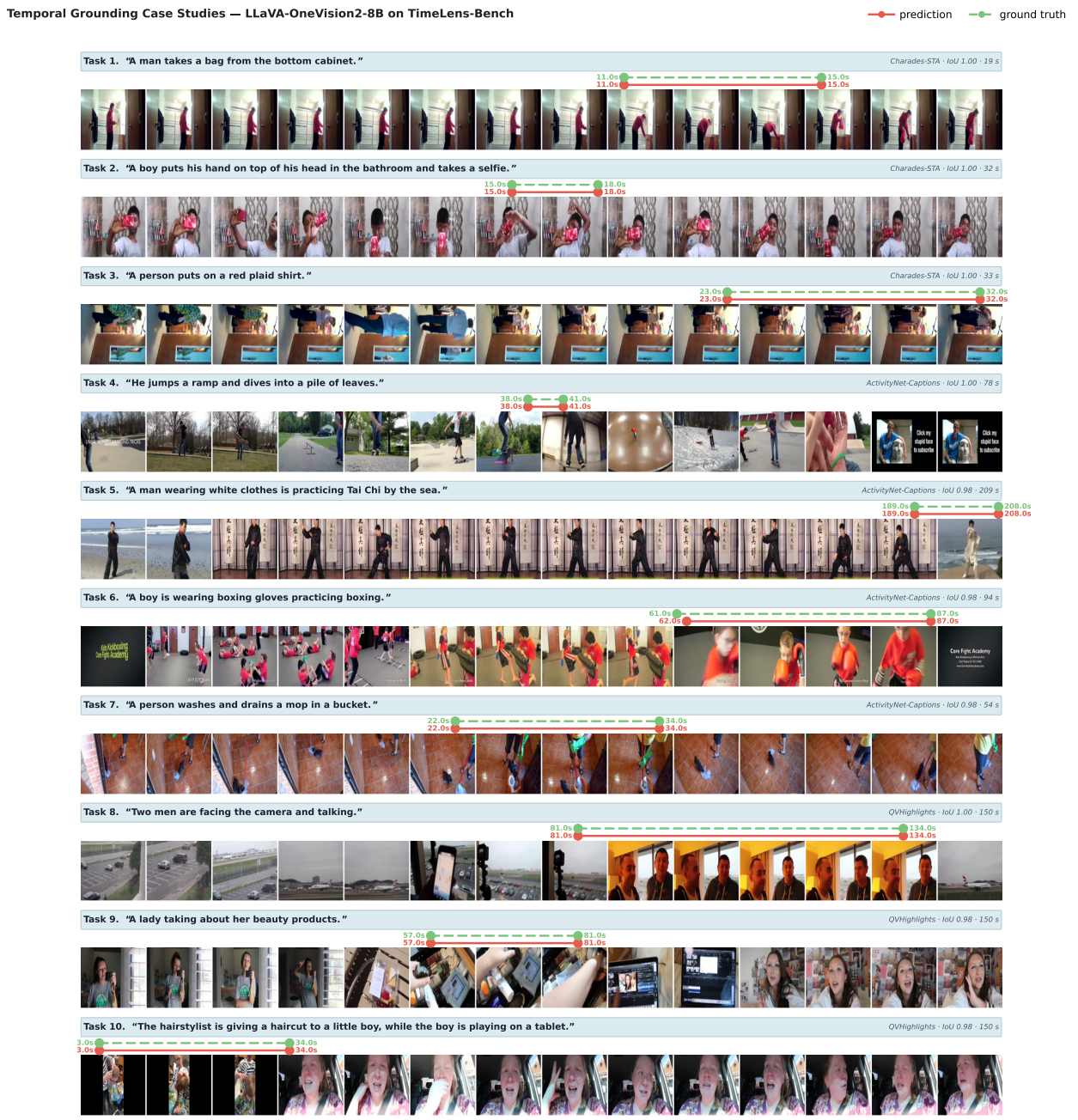


Figure 9 Temporal grounding on TimeLens-Bench. Ten cases from Charades-STA, ActivityNet-Captions, and QVHighlights; per row, mean IoU over five runs.

13.2 Referring Video Object Segmentation and Tracking on ReasonVOS

ReasonVOS evaluates referring video object segmentation given a free-form referring expression and a video. Our model emits a per-frame (x, y) tracking point for the referred object rather than a dense mask directly; the dense mask shown in the figure is produced by feeding these per-frame points to SAM2 (Ravi et al., 2024) downstream as prompts. Figure 10 shows a 36-frame "Track the animal moving forward" case in which a cat moves diagonally across the scene; the point sequence follows the cat through pose change and partial occlusion, yielding $\mathcal{J}\&\mathcal{F} = 0.939$ and HOTA $= 0.954$ on the SAM2 mask.

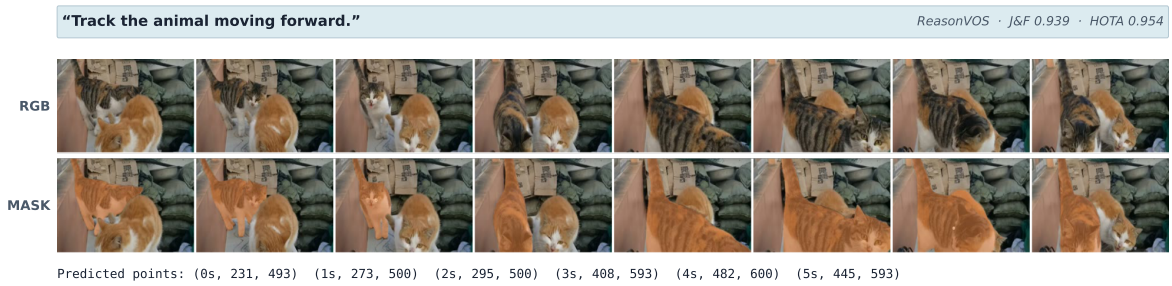


Figure 10 R-VOS on ReasonVOS — "Track the animal moving forward". The two strips show eight evenly-sampled frames of the input RGB (top) and the corresponding SAM2-derived dense mask (bottom) for a 36-frame clip; the per-frame (x, y) tracking points emitted by our model and used as SAM2 prompts are listed below the strips.

13.3 Referring Video Object Segmentation and Tracking on Ref-DAVIS17

Ref-DAVIS17 evaluates the same referring video task on a different visual domain that emphasises high-motion outdoor footage. The pipeline is identical to the ReasonVOS case above: per-frame tracking points are emitted by our model and the dense mask is recovered by SAM2 (Ravi et al., 2024) downstream. Figure 11 shows a 52-frame Drift-Chicane "Track a sport car" case where the point sequence stays aligned through tire smoke, motion blur, and viewpoint change, yielding $\mathcal{J}\&\mathcal{F} = 0.961$ and HOTA $= 0.963$ on the SAM2 mask.

Figure 11 R-VOS on Ref-DAVIS17 — "Track a sport car". Same two-strip layout as Figure 10: the top strip is the input RGB and the bottom strip is the SAM2-derived dense mask, with the model's per-frame (x, y) points printed below.

13.4 Real-World Robot Manipulation

We deploy the model on two tabletop manipulation tasks (apple-to-plate, left; bread-to-oven, right) and re-query it online at three moments during execution. At each query the model receives only the natural-language instruction and the current RGB observation, and returns an image-space list of (x, y, z) waypoints that a downstream IK module follows; the waypoint count contracts as the gripper approaches the target, and the trajectory updates whenever the scene state changes.

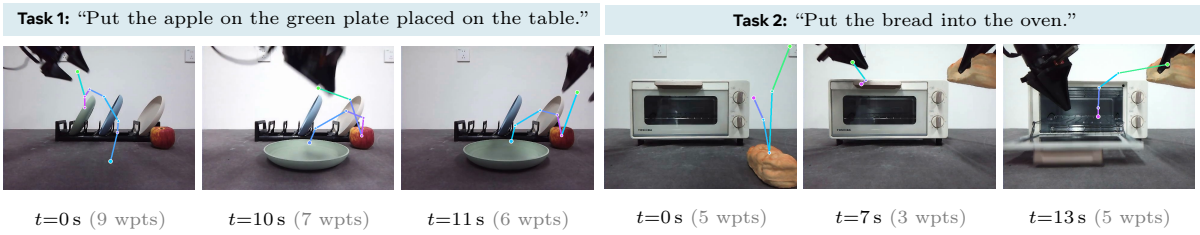


Figure 12 Real-world trajectory predictions on a robot manipulation setup. Cyan polyline = trajectory order; magenta dots = waypoints; green dot = start; z is a normalised depth.

13.5 JumpScore Validation

Figure 13 compares uniform 128-frame sampling and codec-stream sampling on a single JumpScore clip with 85 ground-truth cycle starts at matched visual-token budget. Uniform sampling attributes 14 of 85 cycles correctly (mean IoU 0.116), while codec-stream sampling attributes 82 of 85 (mean IoU 0.894). The improvement reflects codec sampling's ability to concentrate visual tokens on motion-residual regions, exactly where cycle boundaries live in densely repeated jump-rope motion.

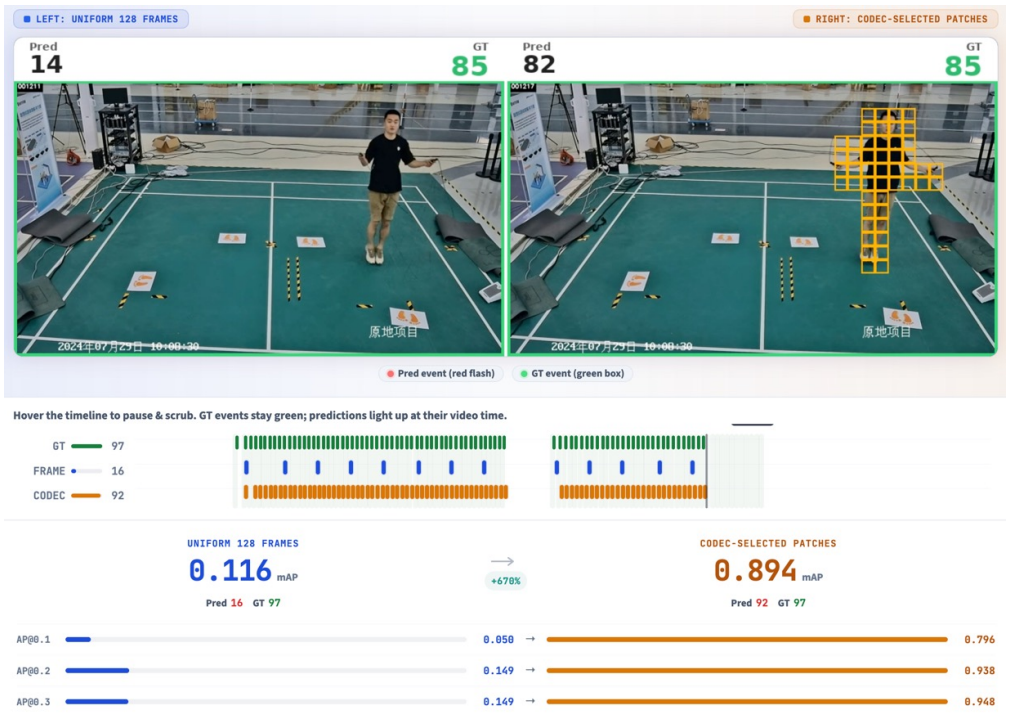


Figure 13 JumpScore validation: uniform vs. codec-stream sampling on a single 85-cycle clip. At matched visual-token budget, codec-stream sampling attributes 82 of 85 cycle starts (mIoU 0.894) versus 14 of 85 for uniform 128-frame sampling (mIoU 0.116); each predicted cycle start is drawn green when it lands within 0.1 s of a ground-truth start and red otherwise.

13.6 2D Spatial Grounding

Figure 14 reports eight 2D pointing cases covering object references, relational queries, and free-space queries; the model emits an (x, y) pixel coordinate (overlaid in red).

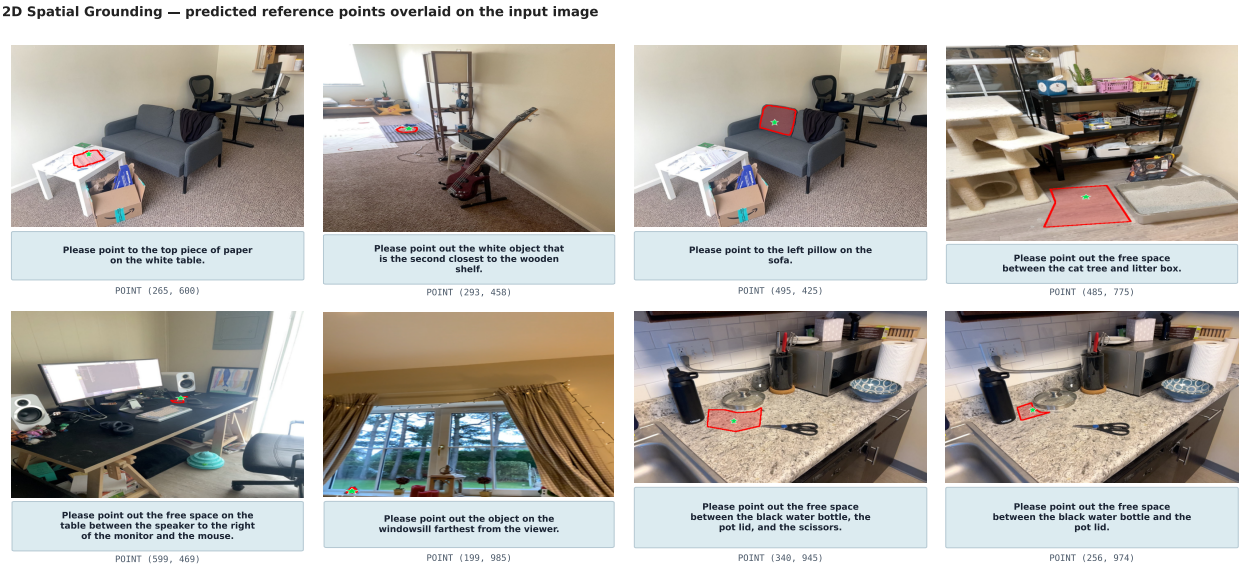


Figure 14 2D spatial grounding. Eight examples; predicted point overlaid in red.

13.7 3D Spatial Grounding

Figure 15 reports five 3D pick-and-place cases ("pick up X, and move it to Y"); the model emits a continuous 3D trajectory in image space with relative depth.

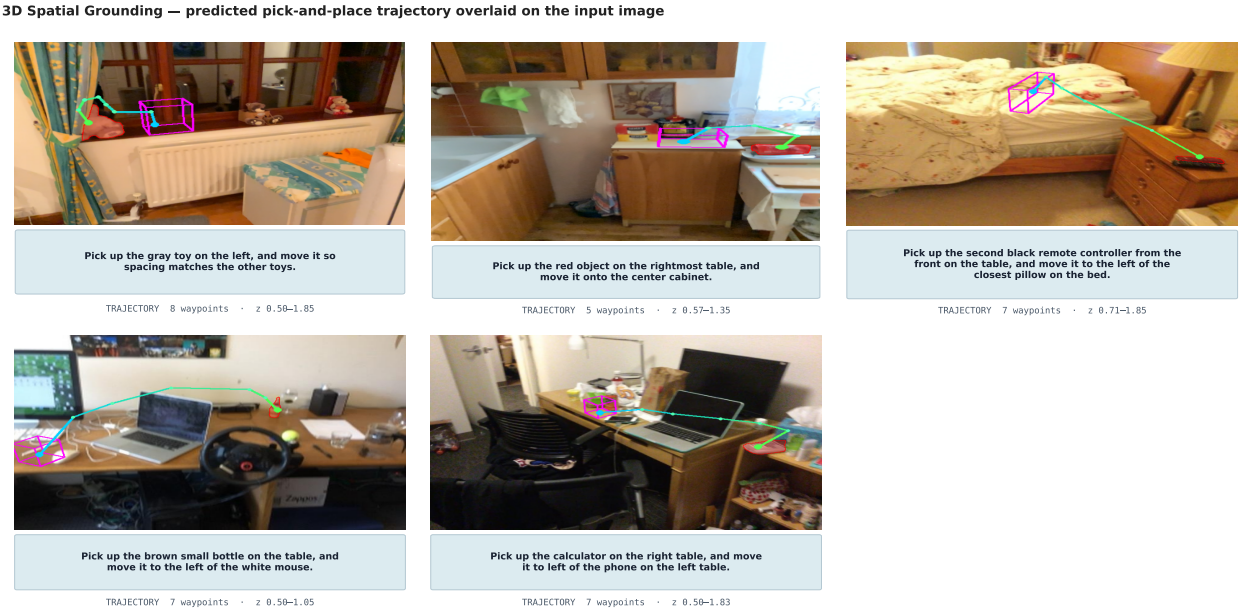


Figure 15 3D spatial grounding. Five pick-and-place examples; predicted 3D trajectory overlaid.

References

Mohamad Alansari et al. SPARROW: Learning spatial precision and temporal referential consistency in pixel-grounded video MLLMs. *arXiv:2603.12382*, 2026.

Allen Institute for AI. Molmo 2: Open weights and open data for pointing, tracking, and spatial reasoning. *arXiv preprint*, 2025.

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, et al. Llava-onevision-1.5: Fully open framework for democratized multimodal training. *arXiv:2509.23661*, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv:2502.13923*, 2025b.

Zechen Bai, Tong Wu, Yueheng Li, Ziwei Liu, Yu Liu, Tao Yu, Jingjing Wang, Liang Lin, Mingsheng Ye, Mike Zheng Shou, et al. One token to seg them all: Language instructed reasoning segmentation in videos. In *NeurIPS*, 2024.

Seunghwan Bang and Hwanjun Song. Reasoning over video: Evaluating how MLLMs extract, integrate, and reconstruct spatiotemporal evidence. *arXiv:2603.13091*, 2026.

Lucas Beyer, Pavel Izmailov, Alexander Kolesnikov, Mathilde Caron, Simon Kornblith, Xiaohua Zhai, Matthias Minderer, Michael Tschannen, Ibrahim Alabdulmohsin, and Filip Pavetic. Flexivit: One model for all patch sizes. In *CVPR*, pages 14496–14506, 2023.

Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In *CVPR*, pages 4599–4603, 2023.

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

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015.

Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. SpatialBot: Precise spatial understanding with vision language models. *arXiv:2406.13642*, 2024.

Houlun Chen et al. Think with grounding: Curriculum reinforced reasoning with video grounding for long video understanding. *arXiv:2602.18702*, 2026a.

Jieneng Chen et al. Thinking with spatial code for physical-world video reasoning. *arXiv:2603.05591*, 2026b.

Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, et al. ShareGPT4Video: Improving video understanding and generation with better captions. *arXiv:2406.04325*, 2024a.

Tao Chen et al. Scaling the long video understanding of multimodal large language models via visual memory mechanism. *arXiv:2603.29252*, 2026c.

Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, et al. LongVILA: Scaling long-context visual language models for long videos. *arXiv:2408.10188*, 2024b.

Yuxiao Chen et al. Learning compact video representations for efficient long-form video understanding in large multimodal models. *arXiv:2602.17869*, 2026d.

An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. SpatialRGPT: Grounded spatial reasoning in vision language models. *arXiv:2406.01584*, 2024.

Zixu Cheng et al. GraphThinker: Reinforcing temporally grounded video reasoning with event graph thinking. *arXiv:2602.17555*, 2026.

Rohan Choudhury, Guanglei Zhu, Sihan Liu, Koichiro Niinuma, Kris Kitani, and László Jeni. Don't look twice: Faster video transformers with run-length tokenization. *NeurIPS*, 37:28127–28149, 2025.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, et al. Molmo2: Open weights and data for vision-language models with video understanding and grounding. *arXiv:2601.10611*, 2026.

Mostafa Dehghani, Basil Mustafa, Josip Djolonga, Jonathan Heek, Matthias Minderer, Mathilde Caron, Andreas Steiner, Joan Puigcerver, Robert Geirhos, Ibrahim M Alabdulmohsin, et al. Patch n'pack: Navit, a vision transformer for any aspect ratio and resolution. *NeurIPS*, 36:2252–2274, 2023.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, Andrew Head, Rose Hendrix, Favyen Bastani, Eli VanderBilt, Nathan Lambert, Yvonne Chou, Arnavi Chheda, Jenna Sparks, Sam Skjonsberg, Michael Schmitz, Aaron Sarnat, Byron Bischoff, Pete Walsh, Chris Newell, Piper Wolters, Tanmay Gupta, Kuo-Hao Zeng, Jon Borchardt, Dirk Groeneveld, Crystal Nam, Sophie Lebrecht, Caitlin Wittlif, Carissa Schoenick, Oscar Michel, Ranjay Krishna, Luca Weihs, Noah A. Smith, Hannaneh Hajishirzi, Ross Girshick, Ali Farhadi, and Aniruddha Kembhavi. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv:2409.17146*, 2024.

Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. MeViS: A large-scale benchmark for video segmentation with motion expressions. In *ICCV*, 2023.

Hao Fei, Shengqiong Wu, Hanwang Zhang, Tat-Seng Chua, and Shuicheng Yan. Vitron: A unified pixel-level vision LLM for understanding, generating, segmenting, editing. In *NeurIPS*, 2024.

Junjie Fei et al. Small vision-language models are smart compressors for long video understanding. *arXiv:2604.08120*, 2026.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou Wang, and Xiangyu Yue. Video-R1: Reinforcing video reasoning in MLLMs. *arXiv:2503.21776*, 2025.

Chaoyou Fu, Haozhi Yuan, et al. Video-MME-v2: Towards the next stage in benchmarks for comprehensive video understanding. *arXiv:2604.05015*, 2026.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*, pages 5267–5275, 2017.

Yiran Guan et al. Video streaming thinking: VideoLLMs can watch and think simultaneously. *arXiv:2603.12262*, 2026.

Yongxin Guo, Jingyu Liu, Mingda Li, Xiaoying Tang, Xu Chen, and Bo Zhao. TRACE: Temporal grounding video LLM via causal event modeling. *arXiv:2410.05643*, 2024a.

Yongxin Guo, Jingyu Liu, Mingda Li, Xiaoying Tang, Xi Liu, and Xu Chen. VTG-LLM: Integrating timestamp knowledge into video LLMs for enhanced video temporal grounding. *arXiv:2405.13382*, 2024b.

Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. VTimeLLM: Empower LLM to grasp video moments. In *CVPR*, 2024a.

De-An Huang, Shijia Liao, Subhashree Radhakrishnan, Hongxu Yin, Pavlo Molchanov, Zhiding Yu, and Jan Kautz. LITA: Language instructed temporal-localization assistant. In *ECCV*, 2024b.

Haoyi Jiang, Liu Liu, Xinjie Wang, Yonghao He, Wei Sui, Zhizhong Su, Wenyu Liu, and Xinggang Wang. Spa3R: Predictive spatial field modeling for 3D visual reasoning. *arXiv:2602.21186*, 2026.

Jindong Jiang, Xiuyu Li, Zhijian Liu, Muyang Ding, Yunsheng Yang, Yu Zhu, Yu Bao, Hongxu Yin, Yao Lu, Song Han, et al. STORM: Token-efficient long video understanding for multimodal LLMs. *arXiv:2503.04130*, 2025.

Woojeong Jin et al. AgentRVOS: Reasoning over object tracks for zero-shot referring video object segmentation. *arXiv:2603.23489*, 2026a.

Xin Jin et al. Compression tells intelligence: Visual coding, visual token technology, and the unification. *arXiv:2601.20742*, 2026b.

Yang Jin, Zhicheng Sun, Kun Xu, Liwei Chen, Hao Jiang, Quzhe Huang, Chengru Song, Yuliang Liu, Di Zhang, Yang Song, et al. Video-lavit: Unified video-language pre-training with decoupled visual-motional tokenization. *arXiv:2402.03161*, 2024.

Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34:11846–11858, 2021.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *TMLR*, 2025a.

Dongxu Li, Yudong Liu, Haoning Wu, Yue Wang, Zhiqi Shen, Bowen Qu, Xinyao Niu, Guoyin Wang, Bei Chen, and Junnan Li. Aria: An open multimodal native mixture-of-experts model. *arXiv:2410.05993*, 2024.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv:2305.06355*, 2023.

Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, et al. VideoChat-Flash: Hierarchical compression for long-context video modeling. *arXiv:2501.00574*, 2025b.

Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. VideoChat-R1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv:2504.06958*, 2025c.

Zhiqi Li, Guo Chen, Shilong Liu, Shihao Wang, Vibashan Vs, Yishen Ji, Shiyi Lan, Hao Zhang, Yilin Zhao, Subhashree Radhakrishnan, et al. Eagle 2: Building post-training data strategies from scratch for frontier vision-language models. *arXiv:2501.14818*, 2025d.

Ming Liu et al. Wan-R1: Verifiable-reinforcement learning for video reasoning. *arXiv:2603.27866*, 2026.

Xiangrui Liu, Yan Shu, Zheng Liu, Ao Li, Yang Tian, and Bo Zhao. Video-XL-Pro: Reconstructive token compression for extremely long video understanding. *arXiv:2503.18478*, 2025.

Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Oryx MLLM: On-demand spatial-temporal understanding at arbitrary resolution. *arXiv:2409.12961*, 2024a.

Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cai, Yuxian Han, Xiuyu Xu, et al. NVILA: Efficient frontier visual language models. *arXiv:2412.04468*, 2024b.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *ACL*, 2024.

Lingchen Meng, Hengduo Li, Bor-Chun Chen, Shiyi Lan, Zuxuan Wu, Yu-Gang Jiang, and Ser-Nam Lim. Adavit: Adaptive vision transformers for efficient image recognition. In *CVPR*, pages 12309–12318, 2022.

Shehan Munasinghe, Hanan Gani, Wenqi Zhu, Hisham Cholakkal, Fahad Shahbaz Khan, Rao M. Anwer, and Salman Khan. VideoGLaMM: A large multimodal model for pixel-level visual grounding in videos. In *CVPR*, 2025.

Kun Ouyang, Yuanxin Liu, Haoning Bai, Yuxin Hu, Lu Hou, Mingxiao Zhou, and Maosong Sun. SpaceR: Reinforcing MLLMs in video spatial reasoning. *arXiv:2504.01805*, 2025.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *NeurIPS*, 34:13937–13949, 2021.

Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad Shahbaz Khan. GLaMM: Pixel grounding large multimodal model. In *CVPR*, 2024.

Hanoona Rasheed, Abdelrahman Shaker, Mahmoud Wajahat, Muhammad Maaz, Tianzhu Hu, Hisham Cholakkal, Salman Khan, and Fahad Shahbaz Khan. VideoMolmo: Spatio-temporal grounding meets pointing. *arXiv:2506.05336*, 2025.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *ICLR*, 2024.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. TimeChat: A time-sensitive multimodal large language model for long video understanding. *arXiv:2312.02051*, 2023.

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:2410.17434*, 2024.

Yujiao Shen et al. A simple baseline for streaming video understanding. *arXiv:2604.02317*, 2026.

Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. Video-xl: Extra-long vision language model for hour-scale video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26160–26169, 2025a.

Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. Video-XL-2: Towards very long-video understanding through task-aware KV sparsification. *arXiv:2506.19225*, 2025b.

Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. RoboSpatial: Teaching spatial understanding to 2D and 3D vision-language models for robotics. *arXiv:2411.16537*, 2024.

Jiafei Song et al. EvoComp: Learning visual token compression for multimodal large language models via semantic-guided evolutionary labeling. *arXiv:2604.17087*, 2026.

Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012.

Feilong Tang, Xiang An, Yunyao Yan, Yin Xie, Bin Qin, Kaicheng Yang, Yifei Shen, Yuanhan Zhang, Chunyuan Li, Shikun Feng, Changrui Chen, Huajie Tan, Ming Hu, Manyuan Zhang, Bo Li, Ziyong Feng, Ziwei Liu, Zongyuan Ge, and Jiankang Deng. OneVision-Encoder: Codec-aligned sparsity as a foundational principle for multimodal intelligence. *arXiv:2602.08683*, 2026.

Xi Tang, Jihao Qiu, Lingxi Xie, Yunjie Tian, Jianbin Jiao, and Qixiang Ye. Adaptive keyframe sampling for long video understanding. In *CVPR*, 2025.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Ziteng Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In *NeurIPS*, 2024.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv:2502.14786*, 2025.

Haibo Wang, Zhiyang Xu, Yu Cheng, Shizhe Diao, Yufan Zhou, Yixin Cao, Qifan Wang, Weifeng Ge, and Lifu Huang. Grounded-VideoLLM: Sharpening fine-grained temporal grounding in video large language models. *arXiv:2410.03290*, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv:2409.12191*, 2024b.

Shihao Wang, Guo Zhang, Zhiqi Li, Shilong Liu, Vibashan Vs, Shiyi Lan, Yiyi Jiang, Junzhong Hu, Devansh Bandyopadhyay, Yishen Ji, et al. VideoITG: Multimodal video understanding with instructed temporal grounding. *arXiv:2507.13353*, 2025a.

Shida Wang et al. Dynamic token compression for efficient video understanding through reinforcement learning. *arXiv:2603.26365*, 2026.

Tai Wang, Xiaohan Mao, Chenming Zhu, Runsen Xu, Ruiyuan Lyu, Peisen Li, Xiao Chen, Wenwei Zhang, Kai Chen, Tianfan Xue, Xihui Liu, Cewu Lu, Dahua Lin, and Jiangmiao Pang. EmbodiedScan: A holistic multi-modal 3D perception suite towards embodied AI. In *CVPR*, 2024c.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, Jingjing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingtong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biqing Qi, Jiaye Ge, Qi Qin, Lu Lu, Zhenxiang Li, Yu Qiao, Yali Wang, Limin Wang, Mingsong Chen,

Wenhai Wang, and Jifeng Dai. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv:2508.18265*, 2025b.

Xiao Wang, Qingyi Si, Jianlong Wu, Shiyu Zhu, Li Cao, and Liqiang Nie. ReTaKe: Reducing temporal and knowledge redundancy for long video understanding. *arXiv:2412.20504*, 2024d.

Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, et al. InternVideo2.5: Empowering video MLLMs with long and rich context modeling. *arXiv:2501.12386*, 2025c.

Ye Wang, Ziheng Wang, Boshen Wang, Yidong Tang, Yongming Yan, Jieyu Sun, Boqian Zhang, Wei Wang, and Yi Wang. Time-R1: Post-training large vision language model for temporal video grounding. *arXiv:2503.13377*, 2025d.

Jiahao Xie et al. SSL-R1: Self-supervised visual reinforcement post-training for multimodal large language models. *arXiv:2604.20705*, 2026.

Mingze Xu, Mingfei Gao, Shiyu Zhou, Jiasen Xiao, Yinglu Niu, Joseph Garcia, Leonid Sigal, Yu Zhang, Bo Pang, Soufiane Belharbi, et al. Slow-fast architecture for video multi-modal large language models. *arXiv:2504.01328*, 2025a.

Weili Xu, Enxin Song, Wenhao Chai, Xuexiang Wen, Tian Ye, and Gaoang Wang. Auroralong: Bringing rnns back to efficient open-ended video understanding. *arXiv:2507.02591*, 2025b.

Xizi Yan, Yumin Xu, Yidong Tang, Yongming Min, Junkai Wen, Mike Zheng Shou, and Joya Chen. TimeRefine: Temporal grounding with time refining video LLM. *arXiv:2412.09601*, 2024.

Yuming Yan, Kai Tang, Sihong Chen, Ke Xu, Dan Hu, Qun Yu, and Pengfei Hu. S-GRPO: Unified post-training for large vision-language models. *arXiv:2604.16557*, 2026.

Biao Yang, Bin Wen, Boyang Ding, et al. Kwai keye-vl 1.5 technical report. *arXiv:2509.01563*, 2025a.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. *arXiv:2412.14171*, 2024.

Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, et al. MMSI-Bench: A benchmark for multi-image spatial intelligence. *arXiv:2505.23764*, 2025b.

Shusheng Yang, Jihan Yang, Ellis Brown, Shengbang Tong, Boyang Liang, Xichen Pan, Ziteng Wang, Adithya Iyer, Sai Charitha Akula, Penghao Wu, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. *arXiv:2511.04670*, 2025c.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. MiniCPM-V: A GPT-4V level MLLM on your phone. *arXiv:2408.01800*, 2024.

Daichi Yashima, Shuhei Kurita, Yusuke Oda, and Komei Sugiura. ReMoRa: Multimodal large language model based on refined motion representation for long-video understanding. In *CVPR*, 2026.

Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *CVPR*, pages 10809–10818, 2022.

Haobo Yuan, Xiangtai Li, Tao Zhang, Zilong Huang, Shilin Xu, Shunping Ji, Yunhai Tong, Lu Qi, Jiashi Feng, and Ming-Hsuan Yang. Sa2VA: Marrying SAM2 with LLaVA for dense grounded understanding of images and videos. *arXiv:2501.04001*, 2025a.

Jiangye Yuan et al. Boosting MLLM spatial reasoning with geometrically referenced 3D scene representations. *arXiv:2603.08592*, 2026.

Liping Yuan, Jiawei Wang, Haomiao Sun, Yuchen Zhang, and Yuan Lin. Tarsier2: Advancing large vision-language models from detailed video description to comprehensive video understanding. *arXiv:2501.07888*, 2025b.

Bowen Zeng et al. HybridKV: Hybrid KV cache compression for efficient multimodal large language model inference. *arXiv:2604.05887*, 2026.

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

Boqiang Zhang, Lei Ke, Ruihan Yang, Qi Gao, Tianyuan Qu, Rossell Chen, Dong Yu, et al. Penguin-vl: Exploring the efficiency limits of vlm with llm-based vision encoders. *arXiv:2603.06569*, 2026a.

Jiahui Zhang, Yurui Chen, Yueming Zhou, Yanpeng Xu, Ziyu Huang, Jilin Mei, Junting Chen, Yu-Jie Yuan, Yong Cai, Hang Zhao, and Li Zhang. From flatland to space: Teaching vision-language models to perceive and reason in 3D. *arXiv:2503.22976*, 2025b.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models. In *ACL*, 2025c.

Le Zhang et al. From where things are to what they are for: Benchmarking spatial-functional intelligence in multimodal LLMs. *arXiv:2605.02130*, 2026b.

Shaolei Zhang, Qingkai Sun, Tian Xie, and Yang Feng. LLaVA-Mini: Efficient image and video large multimodal models with one vision token. *arXiv:2501.03895*, 2025d.

Xiaowen Zhang et al. STVG-R1: Incentivizing instance-level reasoning and grounding in videos via reinforcement learning. *arXiv:2602.11730*, 2026c.

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

Yiming Zhang et al. ReVSI: Rebuilding visual spatial intelligence evaluation for accurate assessment of VLM 3D reasoning. *arXiv:2604.24300*, 2026d.

Zheyu Zhang et al. One token per highly selective frame: Towards extreme compression for long video understanding. *arXiv:2604.14149*, 2026e.

Zijia Zhao, Yuqi Huo, Tongtian Yue, Longteng Guo, Haoyu Lu, Bingning Wang, Weipeng Chen, and Jing Liu. Efficient motion-aware video mllm. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24159–24168, 2025.

Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Zheng, Tiejun Huang, Lu Sheng, and Shanghang Zhang. RoboRefer: Towards spatial referring with reasoning in vision-language models for robotics. *arXiv:2506.04308*, 2025.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Yue Cao, Yangzhou Liu, Weiye Xu, Hao Li, Jiahao Wang, Han Lv, Dengnian Chen, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv:2504.10479*, 2025.

Orr Zohar, Xiaohan Wang, Yann Dubois, Nikhil Mehta, Tong Xiao, Philippe Hansen-Estruch, Licheng Yu, Xiaofang Wang, Felix Juefei-Xu, Ning Zhang, et al. Apollo: An exploration of video understanding in large multimodal models. *arXiv:2412.10360*, 2024.