VideoLLMs Conditioned on Visual Scene Graphs: A Global-Local Approach

Ananth Kalyanasundaram[1] and Ashwanth Krishnan[1]

QpiAI, Bangalore, India
{ananth.k,ashwanth.krishnan}@qpiai.tech

Abstract. Current Video Large Language Models (VideoLLMs) face a fundamental trade-off between correctness and computational feasibility. To manage limited token budgets, existing approaches employ aggressive downsampling, applying as much as 75% compression and systematically discarding fine-grained details like small objects and intricate spatial relationships, while processing at native resolution exceeds memory constraints. We propose visual scene-graph conditioned VideoLLMs that address this dilemma through selective spatio-temporal processing. Rather than encoding entire frames, we leverage pretrained object detection models to extract objects at native resolution, representing each object as a unified descriptor combining CLIP embeddings with spatial coordinates. A masked attention mechanism learns rich spatial features by aggregating information across objects within each frame, with temporal reasoning delegated to the LLM over the sequence of per-frame scene-graph tokens. Through permutation-invariant aggregation, we produce scene-graph tokens concatenated with global context tokens from a frozen Vision Transformer and projected into the language model space. This hybrid global-local architecture is especially effective for static-camera scenarios common in robotics, surveillance, and assistive settings. Our method matches strong baselines on three static-camera benchmarks while using 35% as many tokens as the finetuned Qwen3-VL baseline on average across the evaluation benchmarks, transfers across LLM backbones, and generalizes zero-shot to dynamic-camera temporal reasoning.

Keywords: Video Understanding · Scene Graphs · VideoLLM

1 Introduction

Video understanding has become critical for applications ranging from surveillance and autonomous systems to content moderation and assistive technologies, with the global video analytics market projected to reach $37.4 billion by 2030 [14]. Recent Video Large Language Models (VideoLLMs) [2, 9, 29, 42, 44] achieve impressive results by aligning vision encoders with large language models (LLMs) [1,3,7,25,30], enabling question answering, captioning, and temporal reasoning from visual inputs.

Despite this progress, current VideoLLMs face a fundamental bottleneck in fine-grained understanding. To process minutes-long videos within memory constraints, existing approaches [9,33,44] aggressively downsample frames to nearly

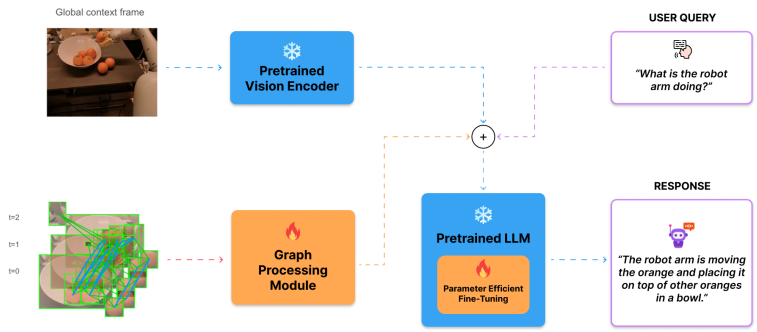

Fig. 1: Overview of our dual branch VideoLLM architecture. The global context branch processes a reference frame through a pretrained vision encoder to capture holistic scene layout. The scene graph branch processes scene graphs made from object detections within and across frames via our Graph Processing Module, which learns soft graph features to generate scene embeddings projected into the tokenizer space. Both token types are concatenated and fed to a pretrained LLM.

240p regardless of the original resolution, discarding spatial details necessary for precise temporal localization and object-level reasoning: distinguishing whether a person *picked up* versus *put down* an object requires hand-object interactions spanning only a few pixels, indistinguishable after downsampling [33].

Recent efforts explore adaptive resolution strategies [4,5,34] or object-centric slots [39], but often remain disjoint from the LLM or rely on "hallucinating" structure via text. Scene graph methods [18], while promising for capturing structure, typically rely on external, non-differentiable parsers that convert visual data into symbolic text, severing gradient flow and losing the rich semantic nuance of the original visual embeddings.

To this end, we propose visual scene graphs as the primary visual modality for VideoLLMs. Analogous to how PointNet [23] reframed 3D perception from voxel grids to permutation-invariant point sets, we reframe video as an unordered set of visual object nodes: each frame is represented as a set of object-level CLIP embeddings [11,24] paired with spatial coordinates, preserving each object's semantic identity independent of frame downsampling. A learnable graph adapter, consisting of a masked attention mechanism, learns soft graph features capturing how objects interact spatially and semantically, without explicit edge annotations or text serialization, and projects them into the LLM's input space as "scene-graph" tokens.

We specifically target **static-camera video understanding**, a regime prevalent in robotics manipulation, surveillance, and assistive technologies, where the constant background makes pixel-level re-encoding of entire frames deeply re-

dundant. Our dual-branch design exploits this directly: a single reference frame in the global stream captures the static layout once, while the scene graph stream tracks how objects move and interact over the full temporal window. Our contributions are:

- An **end-to-end trainable VideoLLM** that processes videos as scene graphs projected directly into the LLM, with a **differentiable masked graph attention adapter** that learns implicit spatial relationships between objects without symbolic edge supervision.
- A **hybrid dual-stream framework** combining fine-grained object graphs with global context, robust to both object-centric interactions and holistic scene changes.
- State-of-the-art parity on BridgeDataV2, HoloAssist, and RoboVQA, three **static-camera** benchmarks, while using only **35% as many tokens as the finetuned Qwen3-VL baseline** on average across the evaluation benchmarks. The adapter is **backbone-agnostic**, transferring from Qwen3-VL to InternVL3.5 [36], and **generalizes zero-shot** to the dynamic-camera TOMATO [27] benchmark, outperforming all baselines on rotation and visual-cue reasoning.

2 Related Work

2.1 Video Language Models

VideoLLMs extend image-language models to the temporal domain by aligning a video encoder with a pretrained LLM. VideoLLaMA [41] and VideoLLaMA 2 [9] use dedicated spatio-temporal connectors to aggregate frame features; LLaVA-Video [44] shows that curated video instructions improve open-ended reasoning. More recently, Qwen2.5-VL [5] and Qwen3-VL [29] introduce dynamic resolution encoding with Multi-Resolution Position Encoding, while Cosmos-Reason1 [2] targets physical reasoning over long-horizon embodied videos.

Despite this progress, these models remain fundamentally *frame-centric*: videos are represented as densely sampled frame patches, aggressively downsampled to fit the LLM's context window. This creates two problems. First, **spatial information loss**: downsampling to 240–320p discards fine-grained details such as small objects and hand-object contact regions [33]. Second, **temporal redundancy**: in static-camera settings, successive frames share nearly identical backgrounds, yet each frame is re-encoded independently, wasting most of the token budget on static context. LongVLM [38] partially addresses redundancy via segment-level hierarchical token merging with global `[CLS]` tokens, but remains patch-based: every segment still re-encodes the full frame, without modeling objects explicitly. In contrast, we encode object crops individually via CLIP [24], decoupling object-level representation from frame-level downsampling, encode the static background *once* via a single reference frame, and devote the remaining budget to a scene graph of objects across time projected as differentiable soft tokens into the LLM, eliminating both failure modes by design.

2.2 Permutation-Invariant Learning

PointNet [23] pioneered permutation-invariant architectures for unordered point clouds using symmetric functions like max pooling, an insight that extends to any unordered collection, including detected objects within frames. Set-LLM [12] extends these principles to pretrained LLMs via prefix masks allowing bidirectional attention within set elements. Like PointNet, we employ max pooling to create permutation-invariant scene embeddings from variable-sized object sets; unlike PointNet, we first apply masked graph attention so objects can exchange information and learn relational structure, a two-stage design in which objects interact through attention while the final scene representation remains invariant to their ordering.

2.3 Scene Graph and Object-Centric Representations

HyperGLM [21] extends pairwise scene graphs with hypergraph structures injected into multimodal LLMs, and ESCA [17] grounds embodied agent perception with a spatio-temporal scene graph aligned via a custom SGCLIP module. ObjectMLLM [28] integrates object bounding boxes and class labels as *symbolic text tokens*, while SAGE [40] constructs State-Action Graphs whose nodes are text embeddings of LLM-generated visual concepts. These methods motivate structured object-level representations, but often express graph nodes or relations in symbolic or language space, which can discard appearance information or place a non-visual intermediate representation between perception and the LLM.

Most closely related to our object-centric formulation, VideoOrion [13] tokenizes object dynamics with a dual-branch VideoLLM: a video-centric branch provides global context tokens, while an object-centric branch uses a detect–segment–track pipeline to maintain object identities across frames and aggregates mask-pooled visual features into compact object tokens encoding spatial-temporal dynamics. This demonstrates that explicitly preserving object-level information can provide an efficient alternative to densely encoding frame patches and is particularly effective for fine-grained object understanding and video-based referring.

Our method shares VideoOrion's motivation of complementing global context with continuous object-level visual features, but differs in how relational and temporal structure are represented. VideoOrion explicitly establishes cross-frame object correspondences and temporally fuses each tracked trajectory into object tokens; in contrast, we construct a detection-driven scene graph independently at each frame, learn soft within-frame relations using masked graph attention, and emit a sequence of permutation-invariant per-frame scene-graph tokens whose temporal dependencies are handled by the LLM's causal attention. Thus, our current formulation does not perform explicit cross-frame tracking; instead, it avoids segmentation and tracking overhead and focuses on differentiable scene-level relational aggregation with a single reference-frame global stream, tailored to the static-camera regime studied here.

3 Method

Given an input video $\mathbf{V} = \{\mathbf{I}_1, ..., \mathbf{I}_T\}$ with T frames and a natural language query $\mathbf{q}$, our goal is to generate a response $\mathbf{r}$ requiring fine-grained spatio-temporal reasoning. As illustrated in Fig. 1, our dual-stream architecture processes both global frame context and object-centric scene graphs.

3.1 Scene Graph Initialization

Object Detection. For each frame $\mathbf{I}_t$, we apply RT-DETR [45], a real-time detection transformer, producing N_t detections $\mathcal{D}_t = \{(\mathbf{b}_i^t, c_i^t, s_i^t)\}_{i=1}^{N_t}$, where $\mathbf{b}_i^t = (x_i, y_i, w_i, h_i)$ is the bounding box, c_i^t the predicted class, and s_i^t the confidence score. We filter detections with $s_i^t < \tau_{\text{det}} = 0.3$.
Visual Feature Extraction. Following recent vision-language models [24,35], we crop each region $\mathbf{b}_i^t$ from frame $\mathbf{I}_t$, resize it to 224×224, and encode it with the CLIP ViT-L/14 image encoder [24], obtaining $\mathbf{v}_i^t \in \mathbb{R}^{1024}$.
Spatial Encoding. Following prior object-centric work [8,10], we encode box geometry as $\mathbf{g}_i^t = \left[\frac{x_i}{W}, \frac{y_i}{H}, \frac{w_i}{W}, \frac{h_i}{H}\right] \in \mathbb{R}^4$, where W and H are the frame width and height; this normalization makes spatial features invariant to frame resolution.
Object Representation. We concatenate visual and spatial features into $\mathbf{o}_i^t = [\mathbf{v}_i^t; \mathbf{g}_i^t] \in \mathbb{R}^{1028}$. For each frame t, we construct a scene graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ with nodes $\mathcal{V}_t = \{\mathbf{o}_1^t, ..., \mathbf{o}_{N_t}^t\}$; unlike traditional scene graphs requiring explicit edge annotations, relationships $\mathcal{E}_t$ are learned implicitly by the module described next.

3.2 Graph Processing Module

The module transforms object-centric representations into coherent scene embeddings via masked graph attention, permutation-invariant aggregation, and temporal sequencing.
Masked Graph Attention. To model interactions between objects within a frame, we introduce a masked multi-head attention mechanism [31] over the object features $\{\mathbf{o}_i^t\}_{i=1}^{N_t}$ that explicitly accounts for the variable number of objects across frames. For frame t, we construct a mask $\mathbf{M}_t \in \{0,1\}^{N_{\max} \times N_{\max}}$, where $N_{\max}$ is the maximum object count across frames, with $\mathbf{M}_t[i,j] = 1$ if $i \leq N_t$ and $j \leq N_t$, and 0 otherwise, ensuring padding objects do not influence attention. With learnable projections $\mathbf{Q} = \mathbf{O}_t\mathbf{W}_Q$, $\mathbf{K} = \mathbf{O}_t\mathbf{W}_K$, $\mathbf{V} = \mathbf{O}_t\mathbf{W}_V$, where $\mathbf{O}_t \in \mathbb{R}^{N_{\max} \times 1028}$ is the padded object feature matrix and $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{1028 \times d_k}$, the masked attention for each head h is

$$\mathbf{A}^h = \text{softmax}\left(\frac{\mathbf{Q}^h(\mathbf{K}^h)^\top}{\sqrt{d_k}}\mathbf{M}_t^{\text{attn}}\right)\mathbf{V}^h$$

where $\mathbf{M}_t^{\text{attn}}[i,j] = -\infty$ if $\mathbf{M}_t[i,j] = 0$, so padding positions receive zero attention weight; head outputs are concatenated and projected. This mechanism

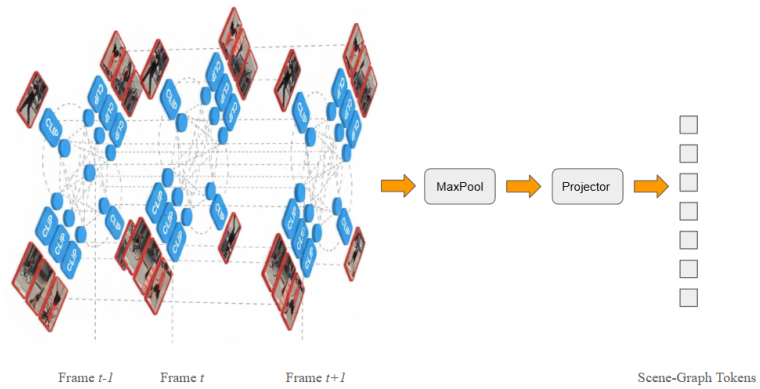

Fig. 2: Graph Processing Module. Objects detected within each frame are processed through masked graph attention (gray dashed connections) to learn spatial relationships. Max pooling aggregates variable-sized object sets into permutation-invariant frame embeddings, projected as per-frame Scene-Graph tokens; the sequence of T tokens is passed to the LLM for temporal reasoning.

lets objects exchange information based on visual and spatial similarity, effectively learning soft graph edges without explicit supervision: spatially proximate objects (*e.g.*, a hand near a cup) attend to each other more strongly. Temporal reasoning is not performed within the attention module; instead, the sequence of T per-frame scene-graph tokens is passed to the LLM, which reasons over object state changes across frames via its causal attention.

Permutation-Invariant Aggregation. Scene understanding should be invariant to detection ordering, which is arbitrary and detector-dependent. We therefore apply max pooling [19,23] across the object dimension, $\mathbf{s}_t = \max_{i=1}^{N_t} \mathbf{O}'_t[i,:] \in \mathbb{R}^{1028}$, setting padding embeddings to $-\infty$ beforehand so they cannot dominate the maximum. Max pooling is chosen over mean pooling or learned aggregation [19] as it is parameter-free, emphasizes distinctive object features, and provides stronger gradients. The scene embeddings are reshaped from $B \times T \times 1028$ to $B \times (T \times 4) \times 257$ to match the spatial merge size of Qwen3-VL's vision encoder, ensuring token-space compatibility with global context tokens.

3.3 Global-Local Feature Combination

While the scene graph stream preserves object-level details, it may miss holistic properties such as lighting, background context, and composition. We therefore add a complementary global context stream processing a reference frame (the first frame) through a standard vision encoder.

Branch Projection and Fusion. Each branch is independently projected into the LLM's token space before fusion. The reference frame $\mathbf{I}$ is processed by the frozen Qwen3-VL ViT encoder [29] and projected via $\mathbf{W}_{\text{global}} \in \mathbb{R}^{d_{\text{ViT}} \times d_{\text{LLM}}}$ to $\mathbf{h}_{\text{global}} = f_{\text{ViT}}(\mathbf{I})\,\mathbf{W}_{\text{global}} \in \mathbb{R}^{B \times N_g \times d_{\text{LLM}}}$; the scene embeddings $\mathbf{S} \in \mathbb{R}^{B \times (T \times 4) \times 257}$

are projected via a separate learned layer $\mathbf{W}_{\text{sg}} \in \mathbb{R}^{257 \times d_{\text{LLM}}}$ to $\mathbf{h}_{\text{local}} = \mathbf{S}\,\mathbf{W}_{\text{sg}}$. Both token sequences are concatenated with the query tokens as $[\,\mathbf{h}_{\mathbf{q}};\, \mathbf{h}_{\text{global}};\, \mathbf{h}_{\text{local}}\,] \in \mathbb{R}^{B \times (L_q + N_g + T \times 4) \times d_{\text{LLM}}}$, semantically aligned in the LLM's input space so the model can selectively attend to global context or fine-grained object details depending on the query. The LLM's causal attention handles both token types without explicit cross-attention modules, with gradients flowing back through $\mathbf{W}_{\text{sg}}$ to the Graph Processing Module.

Training Objective. We finetune end-to-end with the standard language modeling loss: given $(\mathbf{V}, \mathbf{q}, \mathbf{r}^*)$, we minimize $\mathcal{L} = -\sum_{j=1}^{L_r} \log P(r_j^* \mid r_{<j}^*, \mathbf{h}_{\mathbf{q}}, \mathbf{h}_{\text{visual}})$ over the response length L_r. We freeze Qwen3-VL's vision encoder and projection layer to preserve its vision-language alignment, finetune the language decoder with LoRA [16], and train the graph processing module and its projection layer from scratch, allowing the model to learn task-specific graph attention patterns.

4 Experiments

We evaluate our visual scene-graph conditioned VideoLLM on multiple video understanding benchmarks. Ablation studies are presented in Sec. 6.

Training Data. We deliberately select datasets reflecting our static-camera focus, where background redundancy is high and object-level changes carry the primary semantic signal. We perform supervised fine-tuning (SFT) on RoboVQA [26] using captions from the Cosmos-Reason1 SFT dataset [2], which adds diverse video-language instruction pairs enhancing physical reasoning. RoboVQA provides around 218,000 videos (238 hours) across three embodiments (robot, human, human with tool) performing long-horizon instructions; we use a 95:5 train-test split.

Model. We build on the 8B instruction-tuned Qwen3-VL [29], leveraging its pretrained vision encoder and language model; to validate backbone-agnostic transferability, we additionally train a variant on InternVL3.5-8B [36]. Detection uses RT-DETR [45] with a ResNet-101 [15] backbone (108 FPS on a single GPU); object features are extracted with the pretrained CLIP ViT-L/14 encoder [24].

Implementation Details. We train for 3 epochs on 4 NVIDIA A100 (80GB) GPUs with an effective batch size of 32 (micro-batch 1, gradient accumulation 8), completing in roughly 4 days 20 hours, using AdamW with learning rate 2×10^{-5}, cosine scheduling, and 5% warmup. We use LoRA [16] rank 32 for the language model (reducing trainable parameters by 90%), freeze the original vision encoder, and train the graph processing module (4 attention heads, head dimension 257) without LoRA. Object sequences are padded to $N_{\max} = 50$ per frame with attention masks; we sample up to 128 frames uniformly per video during training, though longer sequences are supported at inference. All models train in bfloat16.

5 Results

5.1 Evaluation Benchmarks

We evaluate on three benchmarks using captions from the Cosmos-Reason1 SFT dataset [2]: BridgeDataV2 [32], HoloAssist [37], and the validation split of RoboVQA [26], randomly selecting 1000 videos each from BridgeDataV2 and HoloAssist. All three are captured from static or near-static viewpoints: BridgeDataV2 is table-top robotic manipulation with a fixed camera, HoloAssist captures egocentric human-assistance tasks with largely constant backgrounds, and RoboVQA features mostly fixed-camera robotic embodiments. By encoding the background globally once and tracking objects via the scene graph stream, our architecture exploits this natural redundancy for significant token compression without information loss. To probe generalization beyond this regime, we additionally evaluate zero-shot on TOMATO [27], a dynamic-camera visual temporal reasoning benchmark (Sec. 5.4).

5.2 Qualitative Results

Fig. 3 shows a representative RoboVQA example: a static-camera video of a robotic arm performing an apple-picking task on a cluttered countertop.

Cosmos-Reason1: Center of the frame, occupying most of the lower half, there is an a flat, rectangular surface with a smooth texture and a neutral gray color. It has several drawers underneath. On the table, slightly to the left of center, there is an a large, round bowl with a glossy white finish. It contains several apples and is lined with a piece of white paper. Inside the white bowl and scattered on the table around it, there is a red and yellow apples of varying sizes. They appear fresh and ripe. On the far left side of the table, there is an a rectangular tray containing wrapped candies. The tray is light-colored and contrasts with the darker table surface. In the background, near the top left corner of the frame, there is a two black containers holding utensils like forks and knives. They have a matte finish and are cylindrical in shape.

On the right side of the table, extending towards the apples, there is an a mechanical arm with a metallic finish and multiple joints. It has a gripper at the end designed for picking up objects. The robot arm is positioned over the table, reaching towards an apple. After that, the robot arm's grippers close around an apple located on the table. Then, the robot arm lifts the apple off the table. After this, the robot arm moves upwards and towards the white bowl. Finally, the robot arm releases the apple into the bowl.

Qwen3-VL: In the scene, the following objects are present:

- A white bowl containing several apples.
- Additional apples scattered on the countertop.
- A white rectangular tray with wrapped candies on the left side of the countertop.
- Black containers holding utensils (like spoons or forks) in the background.
- A robotic arm with a gripper mechanism.

Actions of the robot arm:
The robot arm is actively interacting with the apples on the countertop. It picks up one of the apples from the surface and moves it toward the white bowl. The robot then places the apple into the bowl, completing the action of adding it to the existing apples already inside. This suggests the robot is performing a task of organizing or sorting the apples into the bowl.

The robot's movements are precise and controlled, indicating it is likely programmed to handle objects gently and accurately. The scene appears to be set in a kitchen or similar environment, with the robot performing a routine task.

Ours: Center of the frame, occupying most of the lower half, there is an a smooth, gray surface with a modern design, featuring drawers underneath. On the table, slightly to the left of center, there is an a large, white bowl containing several apples, lined with a white paper towel. The bowl contains several apples and is stationary on the table. Inside the bowl and scattered on the table around it, there is a red and yellow apples of varying sizes, some with stickers still attached. On the left side of the table, there is an a rectangular tray containing wrapped candies, neatly arranged. In the background, behind the tray of candies, there is a black containers holding various utensils like forks and knives.

Entering from the right side of the frame, reaching towards the apples, there is an a white robotic arm with articulated joints, designed for precise movements. The robot arm is positioned near an apple on the table, preparing to grasp it. After that, the robot arm closes its grippers and lifts the apple off the table. After that, the robot arm moves upward and slightly to the left, positioning the apple above the bowl. Eventually, the robot arm releases the apple into the bowl.

Fig. 3: Qualitative comparison on a RoboVQA manipulation example. *(Top)* Four sampled frames of a robotic arm performing an apple-picking task. *(Bottom)* Model responses for the prompt: *"What are the objects in the scene and describe the actions of the robot arm in the video?"*

Cosmos-Reason1 (green) produces a detailed, grounded description correctly sequencing the arm's actions (approach → grasp → lift → release into bowl). Qwen3-VL (red) correctly enumerates objects but outputs a structured list rather than a coherent temporal narrative, missing finer details such as the stickers on the apples and the paper lining in the bowl. Our model (blue) matches Cosmos-Reason1's detail, capturing object attributes and the correct temporal sequence while consuming $\sim$**10**$\times$ fewer tokens, demonstrating that object-centric scene graphs preserve the fine-grained spatial and temporal details needed for long-horizon manipulation understanding at a fraction of the token cost.

Fig. 4 shows a further unseen HoloAssist example, where our model again produces descriptions comparable in detail and temporal grounding to Cosmos-Reason1, and Fig. 5 qualitatively illustrates the ablations of Sec. 6: the symbolic baseline fails to capture temporal progressions such as the camera lid being opened and closed, which the visual scene graph stream resolves.

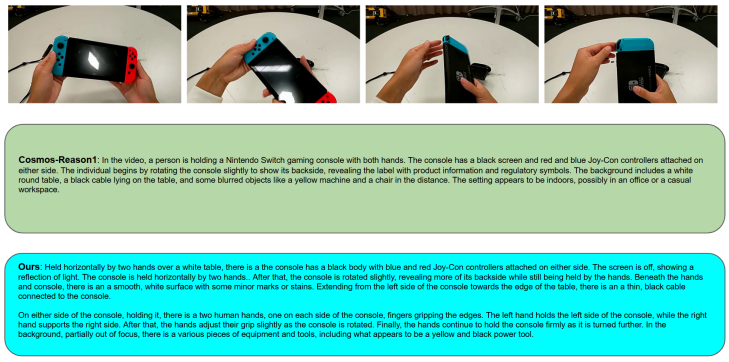

Fig. 4: Qualitative comparison between **Cosmos-Reason1** (green) and **Ours** (cyan) on an unseen example depicting a person manipulating a Nintendo Switch console.

In Fig. 6, while Cosmos-Reason1 correctly sequences the full manipulation (gripping, lifting, unfolding, and releasing the cloth), our model captures the objects and general motion but misses the complete unfolding sequence. This failure mode arises when deformable objects undergo large shape changes across frames that fall outside the detector's localization capability, causing the scene graph stream to lose track of the primary object's state transitions.

5.3 Quantitative Analysis

We compare against Cosmos-Reason1 [2], Cosmos-Reason2-8B, finetuned Qwen3-VL [29] without scene graph conditioning, and InternVL3.5-8B [36], reporting BLEU-4 [22], ROUGE-L [20], METEOR [6], and BERTScore-F1 [43], a complementary stack covering lexical precision, structural recall, paraphrase-aware F-measure, and semantic similarity.

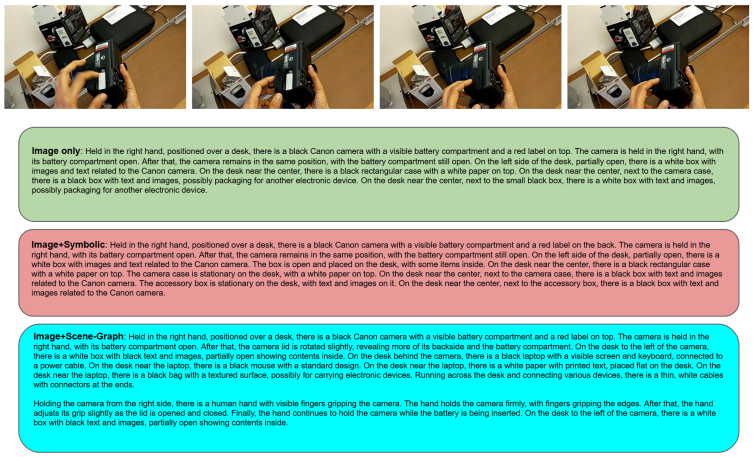

Fig. 5: Qualitative ablation between **Image only** (green), **Image + Symbolic** (red) and **Image + Scene Graph** (cyan) on an unseen example of a hand manipulating a Canon camera, demonstrating the advantage of continuous visual embeddings over both global frame encoding and symbolic text serialization.

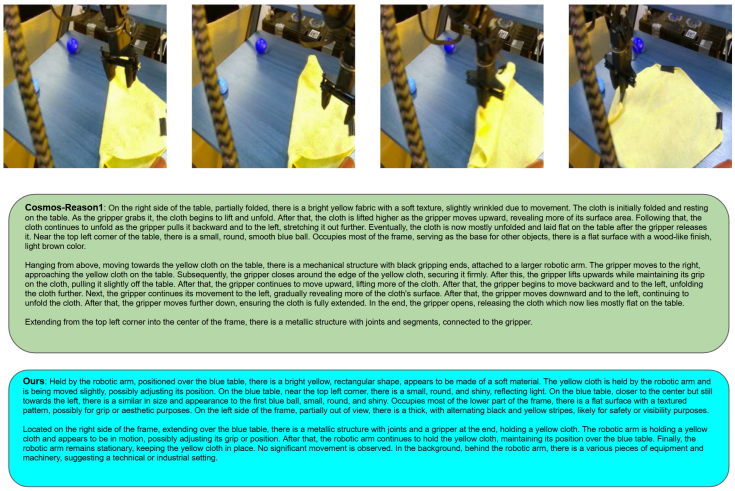

Fig. 6: Failure case: comparison between **Cosmos-Reason1** (green) and **Ours** (cyan) on an unseen BridgeDataV2 example depicting a robotic arm unfolding a yellow cloth on a blue table.

Parity at a Fraction of the Cost. Table 1 shows our Qwen3-VL scene-graph model matches its finetuned base and Cosmos-Reason1 across all three benchmarks, including BridgeDataV2 and HoloAssist, which were entirely *un-*

Table 1: Quantitative comparison on static-camera benchmarks (BLEU-4, ROUGE-L, METEOR, BERTScore-F1). Our scene-graph variants match their fine-tuned base backbones within ±0.3 points at far lower token cost (Table 3); the InternVL3.5 variant substantially outperforms its base.

	BridgeDataV2				**HoloAssist**				**RoboVQA**			
Model	B-4↑	R-L↑	Met↑	F1↑	B-4↑	R-L↑	Met↑	F1↑	B-4↑	R-L↑	Met↑	F1↑
Cosmos-Reason1-7B	10.35	28.90	28.29	86.17	14.37	31.62	31.82	86.91	26.90	44.69	45.25	89.72
Cosmos-Reason2-8B	19.16	38.20	35.75	88.87	21.96	39.85	39.41	88.93	30.58	50.15	46.00	90.65
Qwen3-VL-8B	10.49	28.83	28.49	86.21	14.16	31.43	31.85	86.84	26.73	44.73	45.22	89.70
InternVL3.5-8B	9.34	19.13	13.25	83.42	9.66	18.68	11.26	83.68	12.54	20.86	15.48	84.09
Ours (Qwen3-VL-8B)	10.35	28.80	28.25	86.17	14.11	31.48	**31.91**	86.84	26.88	44.72	45.06	89.71
Ours (InternVL3.5-8B)	8.19	27.34	25.29	84.67	9.70	28.43	27.57	85.68	12.59	31.03	30.36	85.83

seen during training, with differences within ±0.3 points across all four metrics, well within the variance of generation-based evaluation. We argue that *matching this performance at a fraction of the token cost* is itself the primary result, validating our central hypothesis that object-centric scene graphs are significantly more information-dense than frame patches for static-camera video understanding. Notably, our model achieves the highest METEOR among comparable-scale frame-centric baselines, **31.91**, on HoloAssist, whose egocentric fine-grained object manipulations are precisely the regime where object-level representations carry the most discriminative signal. Cosmos-Reason2-8B, trained on substantially larger proprietary physical-reasoning corpora, achieves higher absolute scores; our contribution is orthogonal to such data scaling and can be applied on top of any backbone.

Backbone-Agnostic Transferability. The InternVL3.5 results isolate the adapter's contribution beyond backbone capacity: baseline InternVL3.5-8B scores substantially lower than our InternVL3.5 scene-graph variant across all three datasets (*e.g.*, METEOR 27.57 vs. 11.26 on HoloAssist). Together with the Qwen3-VL parity result, this confirms the architecture transfers across LLM backbones.

5.4 Zero-Shot Generalization to Dynamic Cameras

Our training is scoped to static-camera settings, where background redundancy is highest. To test whether the learned object-centric representations remain useful when this assumption is relaxed, we evaluate zero-shot on TOMATO [27], a visual temporal reasoning benchmark featuring dynamic cameras.

As shown in Table 2, both scene-graph variants outperform all frame-centric baselines, including the smaller Qwen3-VL-4B/2B models, on Rotation and Visual Cues, the categories most dependent on tracking object-level state across frames, while remaining competitive on Shape and Velocity. This indicates the object-centric representation is not merely exploiting static backgrounds but captures transferable object-state dynamics, at drastically lower latency and VRAM (Table 3).

Table 2: Zero-shot accuracy (%) on TOMATO [27]. Without any dynamic-camera finetuning, our scene-graph variants outperform all baselines on Rotation and Visual Cues.

Model	Rotation	Shape	Velocity	Visual Cues
Cosmos-Reason1-7B	14.3	21.5	28.1	40.0
Qwen3-VL-8B	22.7	**38.6**	**31.9**	40.3
Qwen3-VL-4B	17.5	36.3	21.4	38.8
Qwen3-VL-2B	24.1	34.1	25.2	37.3
InternVL3.5-8B	24.1	35.4	22.9	31.4
Ours (Qwen3-VL-8B)	**26.6**	30.0	30.2	**45.7**
Ours (InternVL3.5-8B)	25.5	31.8	31.7	38.6

Table 3: End-to-end inference efficiency on a 100-video long, cluttered RoboVQA stress-test subset. These per-clip measurements are specific to this subset and therefore differ from the benchmark-wide averages reported in Sec. 5.5. Wall-clock time includes RT-DETR detection, CLIP feature extraction, and VLM decoding.

Model	In tok ↓	Out tok	Time (s) ↓	VRAM ↓
Qwen3-VL-8B	20,624	125	9.05	35.33 GB
InternVL3.5-8B	23,200	352	12.97	56.92 GB
Ours (Qwen3-VL-8B)	**1,128**	68	**2.74**	**18.08 GB**
Ours (InternVL3.5-8B)	8,618	85	4.25	19.53 GB

5.5 Efficiency Analysis

While our model achieves parity in accuracy, its decisive advantage is computational efficiency.

Benchmark-Average Token Compression. Averaged over our evaluation benchmarks, Cosmos-Reason1 [2] consumes 12,472 tokens per video and the fine-tuned Qwen3-VL baseline consumes 1,831, whereas our scene-graph approach requires only **633 tokens**. Thus, our model uses **34.6% as many tokens as Qwen3-VL** (a 65.4% reduction) and 5.1% as many as Cosmos-Reason1 (a 94.9% reduction). The corresponding benchmark-wide peak-VRAM averages are **18.71 GB** for ours, 35.60 GB for Qwen3-VL, and 25.10 GB for Cosmos-Reason1. These benchmark-wide averages are distinct from the 100-video long, cluttered RoboVQA stress-test measurements in Table 3.

End-to-End Latency and Clutter Robustness. Token counts alone can hide preprocessing overhead, so Table 3 separately reports end-to-end measurements on a 100-video long, *cluttered* RoboVQA stress-test subset. On this subset, our Qwen3-VL variant uses 1,128 input tokens per clip versus 20,624 for the Qwen3-VL baseline and runs in 2.74 s versus 9.05 s, with 18.08 GB versus 35.33 GB peak VRAM. These values should not be compared numerically with the 633-versus-1,831 benchmark-wide visual-token averages above because they summarize a

Table 4: Ablations on the 500-video unseen subset. Removing the scene graph branch (top) or replacing visual node embeddings with symbolic JSON serialization (middle) degrades METEOR; the symbolic variant additionally inflates token cost 9.4×.

Model	METEOR ↑	Tokens ↓
Global stream only (Image)	29.55	412
Image + Symbolic (JSON-serialized)	29.20	4,777
Image + Scene Graph (Ours)	**30.84**	**507**

different, deliberately long-video subset. Both scene-graph variants are substantially faster than their base models; scene-graph construction overhead is negligible at most **0.3 s per clip**, after which the reduced token count dominates wall-clock savings. Moreover, the hard cap of $N_{\max}$=50 objects per frame bounds scene-graph tokens regardless of clutter: even at maximum occupancy the representation yields far fewer tokens than full frame-patch encoding. This makes our VideoLLM deployable on consumer-grade hardware (*e.g.*, a single RTX 3090 or 4090), whereas the baselines require enterprise-grade A100/H100 GPUs or multi-GPU setups.

6 Ablation Studies

We validate our key architectural choices on a randomly sampled subset of 500 videos, 250 each from BridgeDataV2 and HoloAssist, drawn from the same test splits as our main evaluation. Both datasets were entirely unseen during training, ensuring conclusions reflect generalizable architectural properties; we report METEOR, consistent with our main evaluation.

Contribution of Each Stream. Removing the scene graph branch and relying solely on global image tokens causes a **1.29-point METEOR drop** (Table 4), demonstrating that the scene graph branch contributes discriminative signal beyond holistic frame encoding: the global stream captures static scene layout while the scene graph stream tracks fine-grained object-level changes across frames that global context alone cannot resolve. A qualitative example is provided in Sec. 5.2.

Visual Embeddings vs. Symbolic Representations. A key design choice is using continuous CLIP visual embeddings as soft graph node features rather than serialized symbolic text, the input format employed by symbolic object-centric methods such as ObjectMLLM [28] and SAGE [40]. We compare against a baseline where each frame's detection output is serialized as a structured JSON string, `{"frame": t, "objects": [{"class": ..., "bbox": ...}]}`, injected directly into the LLM's prompt in place of the scene graph token stream, preserving the same structural information (class labels, boxes, confidences) but discarding all visual appearance. Our visual scene graph tokens outperform this baseline by **1.64 METEOR points** while consuming **9.4× fewer tokens** (507

vs. 4,777): symbolic text cannot replicate the appearance information in CLIP embeddings (texture, color, shape, fine-grained attributes), and verbose JSON burdens the context with low-density strings. Crucially, since RT-DETR class predictions are discarded after detection and only box crops are embedded via CLIP, our pipeline is also robust to detector misclassification.

7 Limitations and Future Work

Our training and primary evaluation are scoped to static and near-static cameras; while zero-shot TOMATO results (Sec. 5.4) suggest the learned representations transfer to dynamic settings, fully handling rapid viewpoint changes would require camera motion estimation in the global stream or frame selection. Our approach also depends on the upstream detector: under extreme occlusion, motion blur, or open-vocabulary objects absent from RT-DETR's training set, the scene graph may miss critical entities (a deformable-object failure case appears in Sec. 5.2); joint detector-LLM training or open-vocabulary detectors are natural remedies. Our graph attention models pairwise interactions but not higher-order hyperedges (*e.g.*, person-tool-object triples). Finally, although detection and feature extraction overhead is small ($\leq$0.3 s per clip), it adds constant latency that may matter for ultra-low-latency applications.

8 Conclusion

We presented a VideoLLM architecture that replaces pixel-centric frame encoding with object-centric visual scene graphs as the primary visual modality: each detected object is a continuous CLIP embedding, processed through a differentiable masked graph attention adapter and projected directly into the LLM's token space, eliminating symbolic parsers and preserving end-to-end gradient flow, while a single reference frame captures static background context once. On three static-camera benchmarks spanning robotic manipulation and egocentric assistance, our method matches strong baselines while consuming only 633 tokens on average (95% fewer than Cosmos-Reason1, 65% fewer than Qwen3-VL) with 18.71 GB peak VRAM, enabling consumer-hardware deployment; the architecture transfers across backbones, with the InternVL3.5 variant substantially outperforming its base model, and generalizes zero-shot to the dynamic-camera TOMATO benchmark, outperforming all baselines on rotation and visual-cue reasoning. This validates our core hypothesis that object-centric representations are significantly more information-dense than frame patches. Future work will explore open-vocabulary detection, end-to-end joint training of the detector backbone, and explicit temporal tracklet modeling.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023) 1

2. Azzolini, A., Bai, J., Brandon, H., Cao, J., Chattopadhyay, P., Chen, H., Chu, J., Cui, Y., Diamond, J., Ding, Y., et al.: Cosmos-reason1: From physical common sense to embodied reasoning. arXiv preprint arXiv:2503.15558 (2025) 1, 3, 7, 8, 9, 12
3. Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., et al.: Qwen technical report. arXiv preprint arXiv:2309.16609 (2023) 1
4. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., Ge, W., Guo, Z., Huang, Q., Huang, J., Huang, F., Hui, B., Jiang, S., Li, Z., Li, M., Li, M., Li, K., Lin, Z., Lin, J., Liu, X., Liu, J., Liu, C., Liu, Y., Liu, D., Liu, S., Lu, D., Luo, R., Lv, C., Men, R., Meng, L., Ren, X., Ren, X., Song, S., Sun, Y., Tang, J., Tu, J., Wan, J., Wang, P., Wang, P., Wang, Q., Wang, Y., Xie, T., Xu, Y., Xu, H., Xu, J., Yang, Z., Yang, M., Yang, J., Yang, A., Yu, B., Zhang, F., Zhang, H., Zhang, X., Zheng, B., Zhong, H., Zhou, J., Zhou, F., Zhou, J., Zhu, Y., Zhu, K.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025) 2
5. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025) 2, 3
6. Banerjee, S., Lavie, A.: Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In: Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. pp. 65–72 (2005) 9
7. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020) 1
8. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020) 5
9. Cheng, Z., Leng, S., Zhang, H., Xin, Y., Li, X., Chen, G., Zhu, Y., Zhang, W., Luo, Z., Zhao, D., et al.: Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. arXiv preprint arXiv:2406.07476 (2024) 1, 3
10. Dang, L.H., Le, T.M., Le, V., Tran, T.: Object-centric representation learning for video question answering. In: 2021 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2021) 5
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 2
12. Egressy, B., Stühmer, J.: Set-llm: A permutation-invariant llm. arXiv preprint arXiv:2505.15433 (2025) 4
13. Feng, Y., Li, Y., Zhang, W., Zheng, S., Luo, H., Yue, Z., Lu, Z.: Videoorion: Tokenizing object dynamics in videos. In: 2025 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 20401–20412. IEEE (2025) 4
14. Grand View Research: Video analytics market size, share & trends analysis report. Industry Report (2024) 1
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 7

16. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. ICLR **1**(2), 3 (2022) 7
17. Huang, J., Sethi, A., Kuo, M., Keoliya, M., Velingker, N., Jung, J., Lim, S.N., Li, Z., Naik, M.: Esca: Contextualizing embodied agents via scene-graph generation. arXiv preprint arXiv:2510.15963 (2025) 4
18. Ji, J., Krishna, R., Fei-Fei, L., Niebles, J.C.: Action genome: Actions as compositions of spatio-temporal scene graphs. arXiv preprint arXiv:1912.06992 (2019) 2
19. Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., Teh, Y.W.: Set transformer: A framework for attention-based permutation-invariant neural networks. In: International conference on machine learning. pp. 3744–3753. PMLR (2019) 6
20. Lin, C.Y.: Rouge: A package for automatic evaluation of summaries. In: Text summarization branches out. pp. 74–81 (2004) 9
21. Nguyen, T.T., Nguyen, P., Cothren, J., Yilmaz, A., Luu, K.: Hyperglm: Hypergraph for video scene graph generation and anticipation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 29150–29160 (2025) 4
22. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the Association for Computational Linguistics. pp. 311–318 (2002) 9
23. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017) 2, 4, 6
24. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021) 2, 3, 5, 7
25. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018) 1
26. Sermanet, P., Ding, T., Zhao, J., Xia, F., Dwibedi, D., Gopalakrishnan, K., Chan, C., Dulac-Arnold, G., Maddineni, S., Joshi, N.J., et al.: Robovqa: Multimodal long-horizon reasoning for robotics. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 645–652. IEEE (2024) 7, 8
27. Shangguan, Z., Li, C., Ding, Y., Zheng, Y., Zhao, Y., Fitzgerald, T., Cohan, A.: Tomato: Assessing visual temporal reasoning capabilities in multimodal foundation models. In: International Conference on Learning Representations. vol. 2025, pp. 7593–7734 (2025) 3, 8, 11, 12
28. Tang, Z., Wang, S., Cho, J., Yoo, J., Sun, C.: How can objects help video-language understanding? arXiv preprint arXiv:2504.07454 (2025) 4, 13
29. Team, Q.: Qwen3-vl: Sharper vision, deeper thought, broader action. Technical Report (2025) 1, 3, 6, 7, 9
30. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023) 1
31. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 5
32. Walke, H.R., Black, K., Zhao, T.Z., Vuong, Q., Zheng, C., Hansen-Estruch, P., He, A.W., Myers, V., Kim, M.J., Du, M., et al.: Bridgedata v2: A dataset for robot learning at scale. In: Conference on Robot Learning. pp. 1723–1736. PMLR (2023) 8

33. Wang, H., Xu, Z., Cheng, Y., Diao, S., Zhou, Y., Cao, Y., Wang, Q., Ge, W., Huang, L.: Grounded-videollm: Sharpening fine-grained temporal grounding in video large language models. arXiv preprint arXiv:2410.03290 (2024) 1, 2, 3
34. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., Fan, Y., Dang, K., Du, M., Ren, X., Men, R., Liu, D., Zhou, C., Zhou, J., Lin, J.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024) 2
35. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024) 5
36. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025) 3, 7, 9
37. Wang, X., Kwon, T., Rad, M., Pan, B., Chakraborty, I., Andrist, S., Bohus, D., Feniello, A., Tekin, B., Frujeri, F.V., et al.: Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20270–20281 (2023) 8
38. Weng, Y., Han, M., He, H., Chang, X., Zhuang, B.: Longvlm: Efficient long video understanding via large language models. In: European Conference on Computer Vision. pp. 453–470. Springer (2024) 3
39. Xu, J., Lan, C., Xie, W., Chen, X., Lu, Y.: Slot-vlm: Object-event slots for video-language modeling. Advances in Neural Information Processing Systems **37**, 632–659 (2024) 2
40. Zang, Y., Tang, Z., Cho, J., Yoo, J., Sun, C.: Sage: A unified framework for generalizable object state recognition with state-action graph embedding. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems 4, 13
41. Zhang, H., Li, X., Bing, L.: Video-llama: An instruction-tuned audio-visual language model for video understanding. arXiv preprint arXiv:2306.02858 (2023), https://arxiv.org/abs/2306.02858 3
42. Zhang, P., Zhang, K., Li, B., Zeng, G., Yang, J., Zhang, Y., Wang, Z., Tan, H., Li, C., Liu, Z.: Long context transfer from language to vision. arXiv preprint arXiv:2406.16852 (2024) 1
43. Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: Bertscore: Evaluating text generation with bert. arXiv preprint arXiv:1904.09675 (2019) 9
44. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: Video instruction tuning with synthetic data. arXiv preprint arXiv:2410.02713 (2024) 1, 3
45. Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J.: Detrs beat yolos on real-time object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16965–16974 (2024) 5, 7