TOWARDS SPATIO-TEMPORAL WORLD SCENE GRAPH GENERATION FROM MONOCULAR VIDEOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatio-temporal scene graphs provide a principled representation for modeling evolving object interactions, yet existing methods remain fundamentally frame-centric: they operate primarily in 2D, reason about currently visible objects, and discard entities upon occlusion. We introduce **World Scene Graph Generation (WSGG)**, which constructs a temporally persistent world scene graph encompassing observed and unobserved interacting objects. To support this task, we introduce **ACTIONGENOME4D**, upgrading ActionGenome videos to 4D scenes. We study the task in two settings: supervised WSGG and zero-shot WSGG. For the supervised WSGG setting, we begin by adapting existing VidSGG methods to WSGG to establish strong baselines. Then, we propose a suite of methods (WorldWise, WorldWise+, WorldWise++) that captures joint visual and 3D object representations and treats occlusion as a natural masking signal, reconstructing missing features via camera-pose-conditioned cross-attention over an object's visible history. For zero-shot WSGG, we evaluate open-source MLLMs under strong context-based prompting (including our proposed graph retrieval framework **UWSGG-GRAPHRAG**) strategies with explicit geometric inputs and outputs. Extensive experiments demonstrate the efficacy of the proposed methods.

1 INTRODUCTION

Scene graphs provide a structured, interpretable representation of visual scenes by encoding objects as nodes and their relationships as edges Krishna et al. (2016). Despite significant progress in generating scene graphs from images and video Ji et al. (2019); Cong et al. (2021), the predominant paradigm remains fundamentally *frame-centric*. Current models process frames independently to predict flat, 2D scene graphs that lack both 3D spatial grounding and temporal consistency. Consequently, when an object becomes occluded or exits the camera's field of view, it simply vanishes from the graph. This frame-centric representation (left, Figure 1) fundamentally contrasts with how real-world agents perceive and act.

Developmental psychology has long recognized *object permanence* Spelke (1990) as a foundational prerequisite for physical reasoning: the understanding that objects continue to exist when they leave our direct perception. Achieving such world-centric scene understanding (right, Figure 1) from monocular video requires three capabilities that existing datasets do not jointly provide: (i) 3D spatial grounding of all objects in a shared world coordinate frame, (ii) temporally consistent object identity and tracking across frames, and (iii) dense semantic annotations between all interacting objects, including unobserved objects that are present in the scene but not visible in a given frame.

We aim to bridge this gap with two main contributions: a new task and a new dataset. We formalize the task of **WSGG**, which generalizes conventional VidSGG from frame-centric graphs to temporally persistent, world-anchored scene graphs that account for all interacting objects in the world state. WSGG requires a model to (a) localize all objects via 3D oriented bounding boxes in a shared world frame and (b) predict all pairwise relationships between interacting objects, including those that remain unobserved. We first adapt existing VidSGG methods (STTran Cong et al. (2021), DsgDetr Chen et al. (2019)) to the WSGG setting by augmenting them with a persistent memory buffer and 3D geometric scaffolding. We then propose a family of methods **WORLDWISE, WorldWise+, and WorldWise++**, built on the *Masked Auto-Encoder* (MAE) paradigm: they treat occlusions and camera motion as natural masking events and reframes unseen-object reasoning as a structured completion problem. An associative retriever reconstructs missing object representa-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

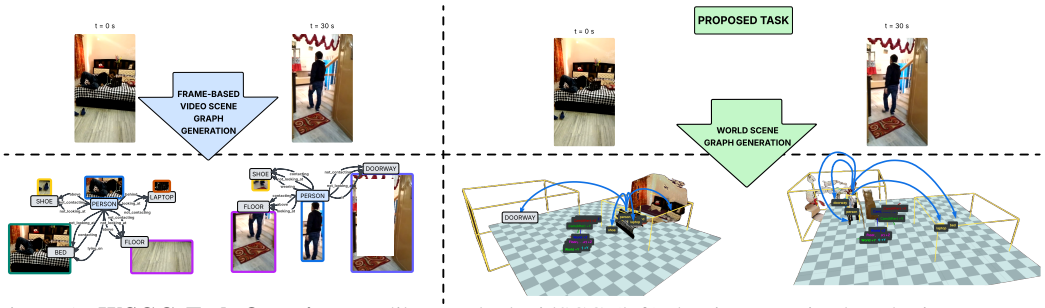


Figure 1: **WSGG Task Overview.** Unlike standard VidSGG (left) that is constrained to the instantaneous camera view and discards objects once they exit the frame or get occluded, our proposed task (right) grounds scene understanding in a global 3D world coordinate frame. WSGG requires a model to output a comprehensive world scene graph at each timestamp containing all interacting objects in the environment. As shown at $t = 30$s, out-of-view objects (e.g., bed, laptop) remain localized in 3D, enabling global, view-independent interpretable scene reasoning. Blue curves drawn from person to objects (right) represent relationships.

tions by attending only to visible temporal observations, while a cross-view reconstruction objective ensures faithful visual feature recovery. To this end, we introduce **ActionGenome4D**, upgrading ActionGenome Ji et al. (2019) into a *4D spatio-temporal scene representation*. We further evaluate open-source MLLMs on zero-shot **U-WSGG** (unlocalized WSGG) and present **UWSGG-GRAPHRAG**, a graph-based retrieval-augmented generation framework. Our contributions are:

1. **ACTIONGENOME4D dataset:** An extension of ActionGenome into 4D, constructed using a human-in-the-loop pipeline to provide 3D oriented bounding boxes (OBBs) for all interacting objects and dense semantic annotations covering unobserved object relationships.
2. **WSGG task:** A novel task that generalizes VidSGG to world-anchored, temporally coherent scene graphs over all interacting objects in the world state from monocular videos.
3. **WORLDWISE-Family:** A novel Masked Autoencoder (MAE)-based framework-based family for WSGG that treats occlusion as masking, along with adapted VidSGG baselines.
4. **UWSGG-GRAPHRAG:** A graph-based Retrieval-Augmented Generation framework for zero-shot U-WSGG that enhances MLLMs' reasoning capabilities.

2 RELATED WORK

Structured Scene Understanding: Scene graph generation has branched into several task variants. Table 1 contrasts these formulations along eight axes[1]. Scene graphs were first introduced as structured representations that encode objects and their pairwise relationships for static images Krishna et al. (2016); Chen et al. (2019); Zhu et al. (2022). Extending scene graphs to the temporal domain requires modeling how relationships evolve across frames. The Video Visual Relation Detection (VidVRD) benchmark Shang et al. (2017) first formalised this task, while the Action Genome (AG) dataset Ji et al. (2019) introduced compositional spatio-temporal scene graphs grounded in human–object interactions. Following AG, several Transformer-based Cong et al. (2021); Feng et al. (2022); Peddi et al. (2025) methods have been proposed to generate dynamic scene graphs for VidSGG.

Departing from the 2D setting, 3D scene graphs aim to encode objects and their relationships in a shared world coordinate frame. These can be thought of as a generalization of 2D scene graphs to the 3D domain. 3D scene graphs were pioneered Kim et al. (2020) as sparse, semantic representations of physical environments for intelligent agents. 3D spatial multimodal knowledge accumulation Feng et al. (2023) advanced scene graph prediction in point clouds. Following this, several works have explored 3D scene graph generation from point clouds Ma et al. (2024); Koch et al. (2023); Zhang et al. (2025a); Nyffeler et al. (2025). Moving forward, 4D scene graphs extend the 3D domain to the temporal dimension, reasoning about objects and their relationships in a shared world coordinate frame across time. 4D panoptic scene graph generation Yang et al. (2024) extends scene graphs to the joint spatial-temporal-panoptic domain. Learning 4D panoptic scene graphs from rich 2D visual scenes was explored in Wu et al. (2025d). RealGraph Lin et al. (2023) contributes a multi-view dataset for context graphs.

[1]Object Scope: *Things*: countable objects; *Stuff*: amorphous regions (e.g., wall, floor).

Table 1: **SGG Tasks. Bold** entries highlight the unique capabilities of WSGG.

Task	Input	Spatial Dim.	Temporal	Localization	Object Scope	Coord. Frame	Unobs. Objects	Rel. Persistence
Image SGG	Image	2D	✗	2D BBox	Detected	Image	✗	Per-frame
Video SGG	Video	2D	✓	2D BBox	Detected / frame	Image	✗	Cross-frame
3D SGG	3D Scan	3D	✗	3D BBox	All in scan	Scene	✗	Per-scan
4D SGG	Video + 3D	3D	✓	3D BBox	Detected / frame	Scene	✗	Cross-frame
Panoptic SGG	Image	2D	✗	2D Mask	Things + Stuff	Image	✗	Per-frame
Panoptic VSGG	Video	2D	✓	2D Mask	Things + Stuff	Image	✗	Cross-frame
WSGG (ours)	**Video**	**3D**	✓	**3D BBox**	**World state**	**World**	✓	**Through occlusion**

Vision-Language Models (VLMs) for Spatial Understanding: VLMs have recently been adapted for spatial understanding tasks that require reasoning about 3D geometry and object relationships from visual input. SpatialVLM Chen et al. (2024) endowed VLMs with spatial reasoning capabilities through internet-scale spatial data collection and training. SpatialRGPT Cheng et al. (2024) grounds spatial reasoning in VLMs through depth-aware region representations. SpatialBot Cai et al. (2025) enables precise spatial understanding with depth integration. SD-VLM Chen et al. (2025b) encodes depth information directly into VLMs for spatial measuring. To further enhance spatial reasoning capabilities of VLMs, several works focus on improving VLM spatial capabilities through specialized training or architectural design Shen et al. (2026); Lee et al. (2025). See Appendix I for an extended related work section and detailed comparison of our work with others.

3 Notation & Problem Description

We present the proposed WSGG task following the formulation of Peddi et al. (2024). Given an input video segment $V_1^T = \{I^t\}_{t=1}^T$ of T monocular frames, we define the *world state* $\mathcal{W}^t$ at timestamp t as the complete set of objects that exist in the scene at that instant. The world state is partitioned into two disjoint subsets: $\mathcal{W}^t = \mathcal{O}^t \cup \mathcal{U}^t, \mathcal{O}^t \cap \mathcal{U}^t = \emptyset$, where $\mathcal{O}^t = \{w_k^t\}_{k=1}^{N(t)}$ is the set of *observed* objects that are visible in frame I^t, and $\mathcal{U}^t = \{w_k^t\}_{k=N(t)+1}^{M(t)}$ is the set of *unobserved* objects that are not visible due to occlusions, camera motion, or other factors. Here $M(t)$ denotes the total number of objects in the world state and $N(t)$ the number of observed objects at timestamp t. We operate under the assumption that all objects, both observed and unobserved, always persist in the world state. We define a binary visibility indicator $\text{vis}(k,t) \in \{0,1\}$ that equals 1 if and only if object $w_k^t \in \mathcal{O}^t$ (i.e., the object is observed in frame I^t). The camera extrinsic parameters at t are given by the SE(3) transformation $\mathbf{T}^t = \begin{bmatrix} \mathbf{R}^t & \boldsymbol{\tau}^t \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} \in SE(3) \subset \mathbb{R}^{4\times4}$, where $\mathbf{R}^t \in SO(3)$ and $\boldsymbol{\tau}^t \in \mathbb{R}^3$ map camera coordinates to world coordinates.

Objects. Let $\mathcal{C}$ denote the set of all object categories. Each object $w_k^t \in \mathcal{W}^t$ is characterised by its category $c_k^t \in \mathcal{C}$, a *3D oriented bounding box* $\mathbf{b}_k^t \in \mathbb{R}^{8\times3}$ given as eight corner vertices in world coordinate frame, and for observed objects, a *2D bounding box* $\mathbf{d}_k^t \in \mathbb{R}^4$ in image coordinates. Because the persistent geometric scaffold provides world-frame grounding at all times, 3D boxes are available for every object in $\mathcal{W}^t$, whereas 2D boxes are defined only for $\mathcal{O}^t$.

Relationships. Let $\mathcal{P}$ be the set comprising all predicate classes. Each pair of objects $\left(w_i^t, w_j^t\right)$ in the world state may exhibit multiple relationships, defined through predicates $\{p_{ijk}^t\}_k$ where $p_{ijk}^t \in \mathcal{P}$. We define a relationship instance r_{ijk}^t as a triplet $\left(w_i^t, p_{ijk}^t, w_j^t\right)$.

Scene Graphs. We consider a scene graph as a symbolic representation of interacting objects and their pair-wise relationships. The *frame-level scene graph* $\mathcal{G}^t$ is the set of all relationship triplets among observed objects: $\mathcal{G}^t = \{r_{ijk}^t \mid w_i^t, w_j^t \in \mathcal{O}^t\}$. The *world scene graph* $\mathcal{G}_\mathcal{W}^t$ is the set of all relationship triplets over the full world state: $\mathcal{G}_\mathcal{W}^t = \{r_{ijk}^t \mid w_i^t, w_j^t \in \mathcal{W}^t\}$. For each object w_i^t and a pair of objects $\left(w_i^t, w_j^t\right)$, we use $\hat{\mathbf{c}}_i^t \in [0,1]^{|\mathcal{C}|}$ and $\hat{\mathbf{p}}_{ij}^t \in [0,1]^{|\mathcal{P}|}$ to represent object-class probabilities and predicate probabilities, respectively. Object probabilities satisfy $\sum_k \hat{c}_{ik}^t = 1$. For ActionGenome4D, only the single-label attention subvector is softmax-normalized: $\sum_{k\in\mathcal{P}_{\text{att}}} \hat{p}_{ijk}^t = 1$. Spatial and contacting predicates use independent sigmoid probabilities for multilabel prediction, with no sum-to-one constraint within either group or across all predicates. Each observed object $w_k^t \in \mathcal{O}^t$ is associated with a visual feature vector $\mathbf{f}_k^t \in \mathbb{R}^{d_{\text{roi}}}$.

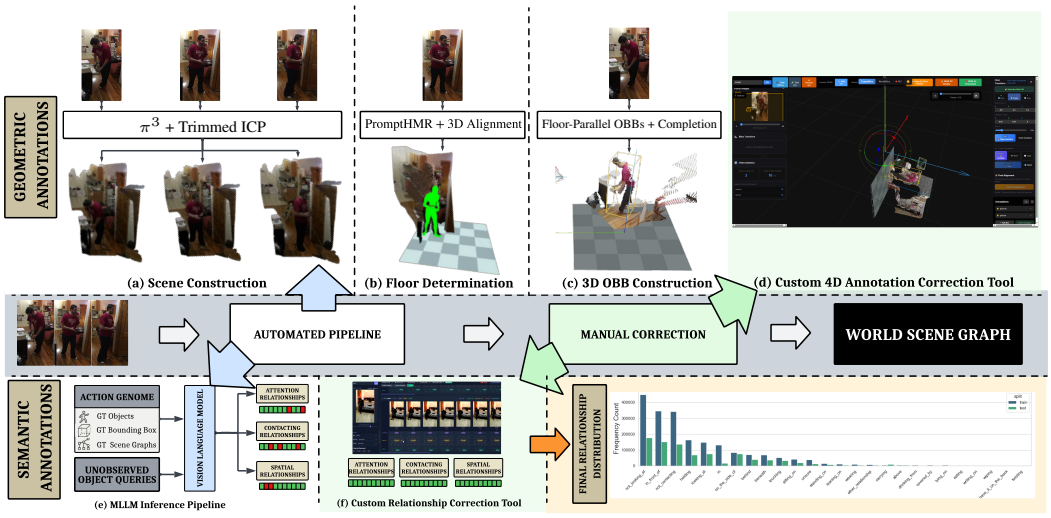


Figure 2: **Hybrid 3D Reconstruction and Annotation.** The framework employs a human-in-the-loop methodology bifurcated into parallel geometric (top) and semantic (bottom) processing streams, progressing from automated initialization to manual refinement. **Geometric Annotations:** Given video sequences, **(a) Scene Construction** uses π^3 reconstruction and trimmed ICP to align dynamic point clouds to a static background. **(b) Floor Determination** utilizes PromptHMR combined with 3D alignment to register human kinematics and robustly anchor a canonical global ground plane. **(c) 3D OBB Construction** then automatically isolates scene objects into floor-parallel 3D Oriented Bounding Boxes (OBBs) using multiscale erosion and temporal completion. **Semantic Annotations:** Concurrently, **(e) the MLLM Inference Pipeline** processes 2D priors from Action Genome (ground-truth objects, bounding boxes, and scene graphs) alongside per-object relationship prompts. An MLLM automates the extraction of relationship predicates, categorizing them into attention, contacting, and spatial relationships. **Manual Correction:** Test annotations are corrected with the **Custom 4D Annotation Correction Tool (d)** for floor alignment, boxes, and temporal trajectories, and the **Custom Relationship Correction Tool (f)** for relationship predicates. Finally, both refined streams are fused to produce the unified **World Scene Graph**, yielding a dataset with a long-tailed distribution of relationships (bottom right).

Problem Description. We formally define the tasks of VidSGG and WSGG as follows: (1) The goal of **VidSGG** is to build frame-level scene graphs $\{\mathcal{G}^t\}_{t=1}^T$ for the observed video segment $V_1^T = \{I^t\}_{t=1}^T$. It entails the detection of observed objects $\{w_k^t\}_{k=1}^{N(t)}$ in each frame and the prediction of all pair-wise relationships $\{r_{ijk}^t\}_{ijk}$ between detected objects. (2) The goal of **WSGG** is to construct the world scene graph $\mathcal{G}_{\mathcal{W}}^t$ at each timestamp t, given the monocular frames V_1^T of a dynamic scene. This entails: (a) Estimating the 3D oriented bounding box $\mathbf{b}_k^t$ in the world coordinate frame for each object $w_k^t \in \mathcal{W}^t$. (b) Predicting all pair-wise relationships $\{r_{ijk}^t\}_{ijk}$ among all interacting objects in $\mathcal{W}^t$, encompassing interactions between all object pairs.

Graph Building Strategies. Following VidSGG conventions, we consider two graph construction strategies: (a) **With Constraint:** Each object pair $\left(w_i^t, w_j^t\right)$ is assigned exactly one predicate p_{ij}^t, yielding a single-edge graph $\mathcal{G}_{\mathcal{W}}^t = \{r_{ij}^t\}_{ij}$. (b) **No Constraint:** Each object pair may carry multiple predicates, producing a multi-edge graph $\mathcal{G}_{\mathcal{W}}^t = \{r_{ijk}^t\}_{ijk}$ that captures richer relational structure.

4 ACTIONGENOME4D DATASET

(a) 3D Scene Construction. We construct per-timestamp 3D scene representations from monocular Action Genome Ji et al. (2019) videos using a feed-forward neural reconstruction model followed by post-hoc geometric alignment. The annotation pipeline (see Figure 2) employs a human-in-the-loop methodology with parallel geometric and semantic processing streams, progressing from automated to manual refinement. Scene construction comprises three stages: (i) frame sampling and preprocessing, (ii) feed-forward 3D inference via the π^3 Wang et al. (2025c) model, and (iii) trimmed iterative closest point (ICP) alignment of dynamic reconstructions to a static background. The output provides the shared world coordinate frame in which every object's 3D oriented bounding box $\mathbf{b}_k^t$ and camera pose $\mathbf{T}^t$ are expressed. Frame sampling uses SIFT matches and a RANSAC homography to estimate overlap with the last retained frame (Appendix C). The π^3 model is a feed-forward neural network for visual geometry reconstruction. A single forward pass jointly estimates per-pixel local 3D coordinates, per-pixel confidence scores, and camera-to-world SE(3) poses $\{\mathbf{T}^t\}_{t=1}^T$ for

all T input views without requiring a fixed reference view. Separate passes reconstruct the original frames and inpainted static views. Trimmed ICP registers each dynamic cloud to the static background using confidence-weighted rigid fitting, and applies the same transform to its camera pose. **Geometric Annotation.** We produce world-frame oriented 3D bounding boxes $\mathbf{b}_k^t \in \mathbb{R}^{8\times 3}$ for every annotated object $w_k^t \in \mathcal{W}^t$ through a multi-stage pipeline encompassing detection, segmentation, floor determination, 3D point extraction, and OBB fitting with temporal completion. GDINO Liu et al. (2024) serves as a detector whose outputs are fused with ground-truth annotations for complete coverage. An LLM classifies each object as static or dynamic. SAM2 Ravi et al. (2024) then produces per-object masks.

(b) Floor Determination. We extract per-object 3D points from the π^3 reconstruction using the segmentation mask and anchor them in a canonical coordinate system. Specifically, we utilize PromptHMR Wang et al. (2025e) to estimate SMPL meshes of the human subject across keyframes. A RANSAC-based similarity estimation aligns the reconstructed point cloud against these meshes, estimating a per-video metric scale and a global ground-plane orientation across valid frames. Stored world coordinates retain reconstruction units; the scale is stored separately for conversion to meters (Appendix D.1.6). **(c) 3D OBB Construction.** Given the floor-aligned 3D points per object, we construct oriented bounding boxes through multiscale erosion (kernels $\{0, 1, \ldots, 10\}$ px), retaining the minimum-volume valid candidate. Automatic boxes use a minimum-area rectangle on the floor plane and vertical extents. Test boxes are refit after manual floor correction using 2D PCA for yaw; both representations remain floor-parallel. Dynamic tracks use linear center/extent interpolation, spherical rotation interpolation, and nearest-frame holding outside observed intervals; static boxes are copied across frames. All boxes are converted to the shared world coordinate frame in which $\mathbf{b}_k^t$ and $\mathbf{T}^t$ are expressed. **(e) Semantic Annotation.** ActionGenome4D extends the original ActionGenome annotations with dense relationship labels for all objects in the world state $\mathcal{W}^t$ on every annotated frame I^t. Thus the predicate set $\mathcal{P}$ spans three disjoint axes: **attention** $\mathcal{P}_{\text{att}}$ (3 labels, single-label), **spatial** $\mathcal{P}_{\text{spa}}$ (6 labels, multi-label), and **contacting** $\mathcal{P}_{\text{con}}$ (17 labels, multi-label). To bootstrap annotations, an MLLM inference pipeline processes 2D priors from ActionGenome alongside per-object relationship prompts for unobserved objects and predicts attention, spatial, and contacting labels for every (person, object) pair on each annotated frame.

(d, f) Manual Correction. Training uses automatically fitted boxes with available floor corrections and pseudo-labels for unobserved relationships. Test annotations undergo manual correction: a purpose-built 3D viewer supports floor alignment, box translation, rotation, scale, and temporal trajectory edits. The relationship interface supports review of each (person, object) pair's attention, spatial, and contacting labels and their temporal extent (see Appendix B.1 for detailed workflow).

Annotation Quality. Our automated pipeline, which integrates π^3 reconstruction, Grounding DINO detection, and SAM2 segmentation, initially achieves a Mean IoU_{3D} of 0.385. While this represents a competitive baseline for zero-shot monocular 3D detection in complex indoor environments, the human-in-the-loop refinement phase via the Custom 4D Annotation Correction Tool (d) raises the inter-annotator agreement measured by 3D IoU to $>$0.9, ensuring metrically accurate and temporally consistent bounding boxes across all frames. We introduce the Label Flip Rate (LFR) as a primary metric to quantify annotation quality; LFR tracks the percentage of VLM-generated predicates that were modified or removed by human annotators. To further validate these labels, we conducted an inter-annotator agreement study on a subset of 200 videos. This yielded high consensus percentages across all semantic axes: 92% for Attention, 90% for Spatial, and 89% for Contacting relationships.

5 EXPERIMENTS

Our experiments are designed to address the following questions: (**RQ1**) What is the efficacy of direct transfer learning of Video Scene Graph Generation models to the WSGG task? (**RQ2**) How effective is adapting and training Video Scene Graph Generation models for the WSGG task? (**RQ3**) How can we design better methods for Supervised WSGG? (**RQ4**) How effective are current Multimodal Large Language Models (MLLMs) in zero-shot WSGG and U-WSGG?

Evaluation Settings and Metrics. Our main supervised evaluation uses the ACTIONGENOME4D test split of 1,511 videos and 48,834 annotated frames. We use two settings. In **PredCls**, during inference, the ground-truth object identities and 3D OBBs are given and methods are tasked to predict the relationships; in **SGDet**, during inference, the methods detect objects and also predict relationships between interacting objects. Following scene graph literature, we report Recall@K

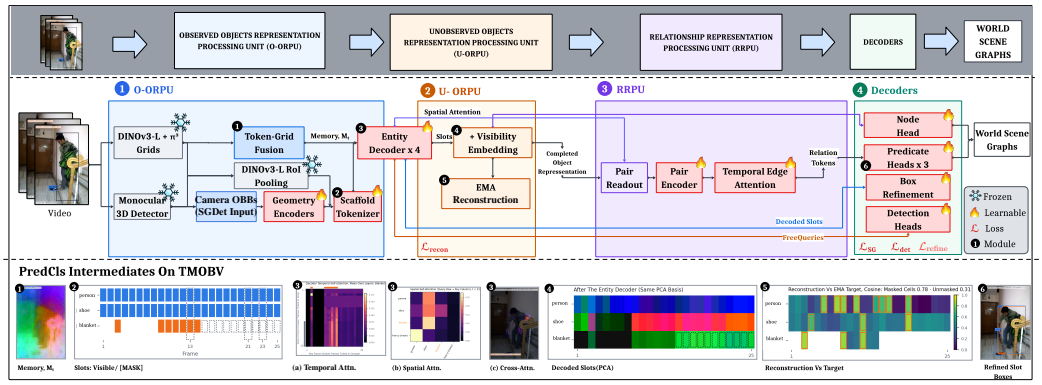


Figure 3: **Overview of the WorldWise++ Architecture.** Top: We show the SGDet setting. The forward pass begins in the O-ORPU, where a frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes supply camera-frame geometry to the graph encoders, and its 2D boxes select the region from which the DINOv3-L RoI appearance of each object is pooled. In parallel, frozen DINOv3-L semantic features and π^3 3D-aware features of each frame are fused to construct the image memory $\mathbf{M}^t$, and geometry encoders compute structural, spatial, object-motion and ego-motion features from the supplied corners and camera poses. The Scaffold Tokenizer fuses these features with the projected DINOv3-L RoI appearance of the object, or with a learned `[MASK]` embedding when the object is out of view or, during training, artificially masked, thus forming one object-query $\mathbf{s}_n^t$ per frame for each object. Next, the Entity Decoder, which is shared by the O-ORPU and the U-ORPU, processes these object-queries and constructs representations for both observed and unobserved objects, giving the decoded object representations $\hat{\mathbf{s}}_n^t$. In the U-ORPU, a visibility embedding is added to distinguish observed objects from constructed unobserved ones, giving the completed object representations $\hat{\mathbf{x}}_n^t$. Next, in the RRPU, a relationship representation is constructed for each person–object pair and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the completed object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\text{SG}}$, $\mathcal{L}_{\text{sim}}$). Bottom: Intermediates of one forward pass of the SGDet model on test video `TM0BV`, where the annotated boxes and object classes take the place of the detector and a tracked blanket leaves the view at frame 15; each panel carries the badge of the module that produced it.

(R@K), the fraction of annotated relationships recovered among the top-K predictions (Lu et al., 2016), and mean Recall@K (mR@K), which averages recall over predicates so that rare predicates count as much as frequent ones (Chen et al., 2019). Both are computed under With Constraint (WC) and No Constraint (NC) graph generation strategies. The supervised WSGG table reports $K = 10$ and $K = 20$; the zero-shot tables report P/R/F1 metrics under PredCls and SGDet settings.

5.1 Supervised WSGG

To address WSGG, we first adapt various existing VidSGG methods[2] for WSGG. We outline the key modifications below and provide the full training details of the adapted VidSGG baselines in Appendix F.2. Next, we propose a suite of models, following the philosophy, each object representation retains its object-query when the object is occluded or leaves the camera view; the model combines temporal object context, scene geometry, and image features to predict its state and relationships. We present and contrast the architecture and training paradigm of our models in-detail in Appendix F.3.2. In this section, we present the proposed method WorldWise++.

WorldWise++ Architecture. It is organised into four processing units (Figure 3): the Observed Objects Representation Processing Unit (O-ORPU), the Unobserved Objects Representation Processing Unit (U-ORPU), the Relationship Representation Processing Unit (RRPU) and the Decoders. Figure 3 shows the forward pass together with its intermediates on a test video.

Observed Objects Representation Processing Unit (O-ORPU). In SGDet, the objects of each frame come from our frozen monocular 3D detector (Appendix F.3.1): its 2D boxes define the regions from which the DINOv3-L tokens of each object are pooled, and its camera-frame 3D boxes are supplied directly to the graph encoders. In PredCls, the annotated boxes and object classes take the place of the detector. The unit receives the object representations with their visibility masks

[2]Direct transfer learning of VidSGG trained models on WSGG, resulted in significantly lower performance.

$\mathrm{vis}(n,t)$, oriented boxes $\mathbf{C}_n^t \in \mathbb{R}^{8\times 3}$ (world-frame annotations in PredCls, camera-frame detector outputs in SGDet), camera poses $\mathbf{T}_t$, and frozen visual features: RoI-pooled DINOv3-L tokens $\mathbf{f}_n^t \in \mathbb{R}^{3072}$ for every visible object, and per-frame DINOv3-L and π^3 token grids reduced to 256 channels by a frozen PCA. Four geometry encoders map the boxes and poses to a structural token $\mathbf{g}_n^t$ (centred corners and centre), a spatial feature $\mathbf{c}_n^t$ (distance and directional components), an object-motion token $\boldsymbol{\mu}_n^t$ (gated finite differences of the supplied box centre) and an ego-motion token $\mathbf{e}_t$ (relative camera pose between frames, encoded over time). Appendix F.1 details the mode-dependent coordinate convention and its limitations. The Scaffold Tokenizer fuses them with the projected appearance into one object-query per object and frame,

$$\mathbf{z}_n^t = \begin{cases} \mathrm{VisProj}(\mathbf{f}_n^t), & u_n^t = 1, \\ \mathbf{z}_{[\mathrm{MASK}]}, & u_n^t = 0, \end{cases} \qquad \mathbf{s}_n^t = \mathrm{LN}\big(\mathrm{ReLU}\big(\mathbf{W}_{\mathrm{fuse}}[\mathbf{g}_n^t \,\|\, \mathbf{z}_n^t \,\|\, \mathbf{c}_n^t \,\|\, \boldsymbol{\mu}_n^t \,\|\, \mathbf{e}_t]\big)\big) \in \mathbb{R}^{256}, \tag{1}$$

where $u_n^t = 1$ marks a usable observation (visible and not artificially masked). Missing appearance, whether caused by natural occlusion or by artificial masks during training, is thus replaced by the same learned mask embedding $\mathbf{z}_{[\mathrm{MASK}]}$ while the geometry is retained. For each frame, the two token grids are normalised and projected to $\mathbf{m}_{\mathrm{dino},xy}^t$ and $\mathbf{m}_{\pi,xy}^t$ at grid cell (x,y) and mixed by a per-cell gate $\mathbf{w}_{xy}^t = \mathrm{softmax}(\mathbf{W}_{\mathrm{gate}}[\mathbf{m}_{\mathrm{dino},xy}^t \,\|\, \mathbf{m}_{\pi,xy}^t])$, giving the image memory $\mathbf{M}_{xy}^t = \mathrm{LN}(w_{xy,1}^t \mathbf{m}_{\mathrm{dino},xy}^t + w_{xy,2}^t \mathbf{m}_{\pi,xy}^t) + \mathbf{P}_{\mathrm{2D}}(x,y)$ with a two-dimensional sine positional encoding.

Entity Decoder (shared by the O-ORPU and the U-ORPU). Per frame, the Entity Decoder processes $Q{=}30$ learned free queries, which are used for detection, followed by the N object-queries $\mathbf{s}_n^t$, whose positional embedding is a projection of the structural token. Each of its four layers (8 heads, feed-forward width 1,024) updates the object-queries and free queries through (i) temporal self-attention over the frames of each object, with learned frame positions, (ii) spatial self-attention among all object-queries and free queries of a frame, whose per-head weights $\mathbf{A}^{t,\ell}$ are kept for the Pair Readout, (iii) cross-attention to the image memory $\mathbf{M}^t$, and (iv) a feed-forward network. The model here performs an offline world-state completion. Cross-attention grounds every object-query in the whole frame, and is the source from which the free queries detect. A final LayerNorm gives the decoded object representations $\hat{\mathbf{s}}_n^t$ and the decoded free queries.

Unobserved Objects Representation Processing Unit (U-ORPU). A two-state visibility embedding, indexed by u_n^t, is added to the decoded object representations $\hat{\mathbf{s}}_n^t$, distinguishing observed objects from constructed unobserved ones and giving the completed object representations $\hat{\mathbf{x}}_n^t$. A linear reconstruction head $\mathbf{W}_{\mathrm{rec}}$ maps $\hat{\mathbf{x}}_n^t$ back to the projected visual space, where it is trained against an exponential moving-average (EMA) copy of the visual projector.

Relationship Representation Processing Unit (RRPU). For each person–object pair $k = (i,j)$, where i indexes the person (the subject) and j the object, the Pair Readout combines the Entity Decoder's spatial attention weights in both directions from every layer and head with the elementwise product of the two decoded object representations, in the manner of EGTR Im et al. (2024):

$$\mathbf{v}_k^t = \mathrm{LN}\big(\mathrm{ReLU}(\mathbf{W}_{\mathrm{pair}}[\boldsymbol{\alpha}_k^t \,\|\, \hat{\mathbf{s}}_i^t \odot \hat{\mathbf{s}}_j^t])\big), \qquad \boldsymbol{\alpha}_k^t = \big[\mathbf{A}_h^{t,\ell}(i,j),\, \mathbf{A}_h^{t,\ell}(j,i)\big]_{\ell\le 4,\, h\le 8} \in \mathbb{R}^{64}. \tag{2}$$

The Pair Encoder then forms the relationship representation from the two completed object representations, the pair readout, the projected CLIP embeddings $\mathbf{t}_c$ of the two object classes and the projected pair geometry $\boldsymbol{\gamma}_k^t$ (direction, log-distance, log-volumes and minimum corner distances),

$$\mathbf{r}_k^t = \mathrm{LN}(\mathrm{ReLU}(\mathbf{W}_{\mathrm{in}}[\hat{\mathbf{x}}_i^t \,\|\, \hat{\mathbf{x}}_j^t \,\|\, \mathbf{v}_k^t \,\|\, \mathbf{t}_{c(i)} \,\|\, \mathbf{t}_{c(j)} \,\|\, \boldsymbol{\gamma}_k^t])) \in \mathbb{R}^{256}. \tag{3}$$

The relationship representations pass through layers of intra-frame attention among the pairs of a frame, and temporal edge attention then refines them over its pair's sequence across the video.

Decoders (Prediction Heads). The node head classifies the completed object representations $\hat{\mathbf{x}}_n^t$, and three predicate heads map each refined relationship representation to attention logits $\hat{\mathbf{y}}_k^{\mathrm{att}}$ (3 classes, softmax), spatial logits $\hat{\mathbf{y}}_k^{\mathrm{spa}}$ (6 labels, sigmoid) and contacting logits $\hat{\mathbf{y}}_k^{\mathrm{con}}$ (17 labels, sigmoid). Two zero-initialised residual heads refine the supplied geometry of every object from its decoded object representation, $\hat{\mathbf{B}}_n^t = \mathbf{B}_n^t + \mathrm{MLP}_{\mathrm{box}}(\hat{\mathbf{s}}_n^t)$ for the normalised 2D box and $\hat{\mathbf{C}}_n^t = \mathbf{C}_n^t + \mathrm{MLP}_{\mathrm{corner}}(\hat{\mathbf{s}}_n^t)$ for the 3D corners, so the untrained model returns its input geometry. The detection heads map the decoded free queries to object/no-object labels, 2D boxes, and 3D corners through the factorised dimension, yaw, depth and centre-offset parameterisation.

Table 2: **Supervised WSGG SGDet (left) and PredCls (right).**

	SGDet												PredCls											
	With Constraint				No Constraint				UO (No Constraint)				With Constraint				No Constraint				UO (No Constraint)			
Method	**R**		**mR**		**R**		**mR**		**R**		**mR**		**R**		**mR**		**R**		**mR**		**R**		**mR**	
	@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
W-STTran (Cong et al., 2021)	12.6	23.6	4.7	12.2	15.6	28.3	8.9	18.1	**11.3**	21.4	2.7	6.0	62.5	68.2	30.9	37.6	74.6	92.6	44.8	70.0	53.0	80.4	13.4	36.4
W-STTran++ (Cong et al., 2021)	12.4	23.4	4.4	11.2	14.7	27.8	8.0	16.4	**11.3**	**22.1**	2.5	6.2	61.3	66.9	28.0	34.0	73.8	92.4	41.0	63.6	56.0	82.9	16.2	34.5
W-DSGDetr (Feng et al., 2022)	13.0	24.2	4.8	12.2	16.9	29.5	9.2	18.3	9.5	20.0	2.4	5.8	62.0	67.5	27.9	33.4	73.8	92.2	40.5	65.6	**57.2**	82.8	13.9	32.8
W-DSGDetr++ (Feng et al., 2022)	13.3	24.3	4.8	12.8	17.1	30.0	10.9	20.5	8.6	20.0	2.3	6.6	62.8	68.5	32.1	38.9	74.7	92.7	47.0	71.0	53.3	79.9	12.4	32.2
W-USG (Wu et al., 2025c)	12.7	23.8	4.7	12.4	15.8	28.0	7.7	16.3	9.5	20.2	2.2	5.7	61.6	67.3	27.5	33.4	74.0	92.6	39.8	64.0	55.1	**83.8**	14.7	34.6
WorldWise	15.0	27.3	5.9	14.5	20.2	32.4	16.5	26.5	5.2	12.8	3.3	7.7	63.2	69.0	40.1	49.6	74.4	92.6	60.1	82.4	52.9	81.6	21.4	44.8
WorldWise+	16.2	28.9	6.3	15.7	**21.3**	33.9	**19.7**	30.5	5.1	12.1	3.2	10.1	67.6	73.5	**44.3**	**54.7**	77.7	94.3	**65.5**	85.6	52.2	82.5	25.8	48.5
WorldWise++	**16.5**	**29.5**	**6.6**	**16.7**	21.1	**34.5**	19.3	**30.9**	6.1	14.1	**4.6**	**11.8**	**68.8**	**74.8**	43.9	54.4	**78.5**	**94.6**	63.8	**86.2**	56.5	83.5	**27.4**	**51.3**
Δ vs. best baseline (%)	+24.1	+21.4	+37.5	+30.5	+23.4	+15.0	+77.1	+50.7	−46.0	−36.2	+70.4	+78.8	+9.6	+9.2	+36.8	+39.8	+5.1	+2.0	+35.7	+21.4	−1.2	−0.4	+69.1	+40.9

Training. We artificially mask visible object appearances with probability 0.3. Their original relationship labels supervise the predictions made from their completed object representations, and their held-out visual features provide reconstruction targets.

Learning objective. Let $\mathcal{P}$ be the valid person–object pairs of a video, $N_{\mathcal{P}} = |\mathcal{P}|$, $\mathcal{V} \subseteq \mathcal{P}$ the pairs whose object is visible, $\mathcal{A} \subseteq \mathcal{V}$ those whose object was artificially masked, and $\ell_k = \mathrm{CE}(\hat{\mathbf{y}}_k^{\mathrm{att}}, y_k^{\mathrm{att}}) + \mathrm{BCE}(\hat{\mathbf{y}}_k^{\mathrm{spa}}, \mathbf{y}_k^{\mathrm{spa}}) + \mathrm{BCE}(\hat{\mathbf{y}}_k^{\mathrm{con}}, \mathbf{y}_k^{\mathrm{con}})$ the predicate loss of pair k. Training combines relationship supervision on visible pairs ($\mathcal{L}_{\mathrm{SG}}$) and on artificially masked pairs ($\mathcal{L}_{\mathrm{sim}}$), visual-feature reconstruction, free-query detection, and box refinement:

$$\mathcal{L}_{\mathrm{WorldWise++}} = \underbrace{\mathcal{L}_{\mathrm{SG}} + \mathcal{L}_{\mathrm{sim}} + \lambda_{\mathrm{recon}}\lambda_{\mathrm{dom}}\mathcal{L}_{\mathrm{recon}}}_{\mathcal{L}_{\mathrm{completion}}} + \lambda_{\mathrm{det}}\mathcal{L}_{\mathrm{det}} + \lambda_{\mathrm{refine}}\mathcal{L}_{\mathrm{refine}}, \tag{4}$$

$$\mathcal{L}_{\mathrm{SG}} = \frac{1}{N_{\mathcal{P}}} \sum_{k \in \mathcal{V} \setminus \mathcal{A}} \ell_k, \qquad \mathcal{L}_{\mathrm{sim}} = \frac{1}{N_{\mathcal{P}}} \sum_{k \in \mathcal{A}} \ell_k,$$

$$\mathcal{L}_{\mathrm{recon}} = \frac{1}{|\mathcal{A}_{\mathrm{cells}}|} \sum_{(t,n) \in \mathcal{A}_{\mathrm{cells}}} \left\| \mathbf{W}_{\mathrm{rec}} \hat{\mathbf{x}}_n^t - \mathrm{VisProj}_{\bar{\theta}}(\mathbf{f}_n^t) \right\|_2^2.$$

$\mathcal{L}_{\mathrm{sim}}$ has the form and denominator of $\mathcal{L}_{\mathrm{SG}}$: it partitions the visible bucket rather than adding a weighted term, and it supplies supervised examples of missing appearance without a pseudo-label for any naturally unobserved pair. Naturally unobserved pairs therefore receive no edge supervision but are counted in $N_{\mathcal{P}}$. Reconstruction minimises squared error at the artificially masked cells $\mathcal{A}_{\mathrm{cells}}$ against a detached target from the EMA projector, $\bar{\theta} \leftarrow m\bar{\theta} + (1-m)\theta$ with $m = 0.996$. To address predicate imbalance, $\mathcal{L}_{\mathrm{SG}}$ and $\mathcal{L}_{\mathrm{sim}}$ use training-time logit adjustment Menon et al. (2021): the offsets $\tau \log \pi_c$ (attention) and $\tau \log \frac{\pi_c}{1-\pi_c}$ (spatial, contacting), with training predicate priors π_c and $\tau = 0.5$, are added to the logits in training, and inference uses unadjusted logits. Detection matches the free queries of each frame to its visible objects with the Hungarian algorithm, using the cost $-p_{\mathrm{cls}} + 5\|\hat{\mathbf{b}} - \mathbf{b}\|_1 + \frac{1}{24}\|\hat{\mathbf{C}} - \mathbf{C}\|_1$, and assigns every unmatched query with box IoU above 0.5 to a target as a one-to-many auxiliary match; $\mathcal{L}_{\mathrm{det}}$ is the class cross-entropy over all queries (no-object weight 0.1) plus the ℓ_1 box loss (weight 5) and the corner loss over the matches.

Optimization and ablation. We train only the graph model, from random initialisation. We use AdamW with learning rate and weight decay both 10^{-4}, a linear warm-up over the first 10% of steps followed by cosine annealing, gradient clipping at 5.0, mixed precision, and a configured budget of 20 epochs. The loss weights are $\lambda_{\mathrm{recon}} = 0.5$, $\lambda_{\mathrm{dom}} = 0.1$, and $\lambda_{\mathrm{det}} = \lambda_{\mathrm{refine}} = 1$.

Insights: WorldWise++ consistently surpasses adapted baselines in aggregate recall and especially class-balanced mean recall, an advantage driven by balanced predicate prediction and superior detection that persists even under a matched detector. Furthermore, while raw recall on unobserved objects is bounded by proposal misses and skewed by dominant head predicates, WorldWise++ maintains a decisive lead in unobserved mean recall, with ablations confirming that detection queries and supervision are critical for safeguarding relationships beyond the visible frame.

5.2 ZERO-SHOT U-WSGG

While the full WSGG task requires 3D-localised detection and tracking (addressed by WorldWise), the *unlocalized* variant (U-WSGG) isolates the core *semantic reasoning* challenge: given a known set of objects, predict their relationships to the person at each frame across attention, spatial, and contacting axes. Modern VLMs can be directly prompted for relationship prediction, but vanilla zero-shot querying suffers from *attention dilution* over long video contexts and the lack of *query-specific temporal grounding*. We propose **UWSGG-GRAPHRAG**, a Graph-structured Retrieval-Augmented Generation strategy (Figure 4) that constructs a video-level knowledge graph offline and uses it as a structured retrieval index, enabling query-specific context selection before the VLM call.

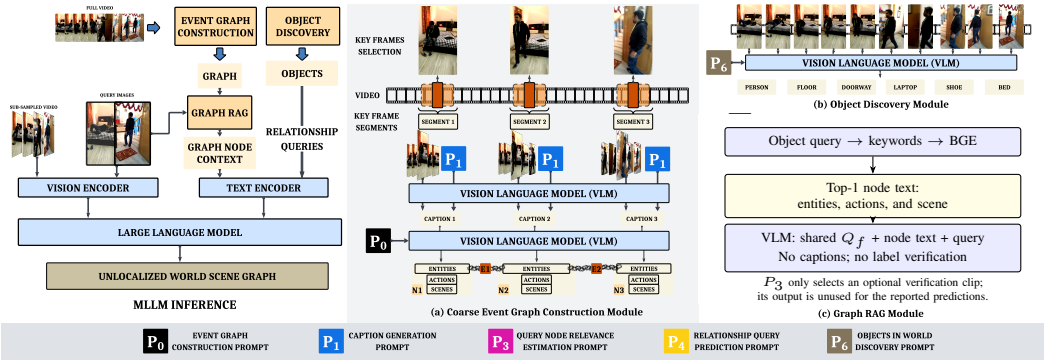


Figure 4: **UWSGG-GRAPHRAG. (a) Coarse Event Graph Construction:** A VLM captions key-frame segments (P_1), then extracts entities, actions and scenes (P_0); an entity index links event nodes sharing an entity key. **(b) Object Discovery:** In SGDet, the VLM discovers objects from whole-video frames and captions (P_6). **(c) Graph RAG:** BGE embeddings retrieve nodes once per object; the top-1 node supplies context at every target frame, reusing retrieval across $|\mathcal{F}|$ annotated frames. The node check (P_3) only selects an optional verification clip and does not affect reported predictions. **Inference:** The relationship prompt receives graph context without captions and the same target-frame-plus-context tensor Q_f as caption prompting. Label verification (P_5) is disabled in reported runs. Appendix H.1.2 specifies the evaluated path through the schematic.

Table 3: **UWSGG-GRAPHRAG: SGDet (left) and PredCls (right) results.** Per-axis precision (P), recall (R), and F1, with micro-F1 (μF1) and macro-F1 (MF1), under the object-set evaluation protocol. Scores are percentages; the Δ rows give the relative change (%) from Caption to RAG within each backbone.

		SGDet											PredCls									
Model	**Method**	**Attention**			**Contacting**			**Spatial**			**μF1**	**MF1**	**Attention**			**Contacting**			**Spatial**			**μF1**	**MF1**
		P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**		
InternVL2.5	Caption	39.9	23.5	**29.6**	39.2	22.4	28.5	21.5	10.5	14.1	23.8	9.8	38.9	38.9	38.9	35.4	31.3	33.2	19.3	15.2	17.0	29.3	17.7
	RAG	38.5	23.2	29.0	39.0	24.6	**30.2**	21.1	10.6	14.1	**24.2**	**11.0**	41.8	41.8	**41.8**	35.6	33.8	**34.7**	26.3	21.0	**23.4**	**32.9**	**18.8**
	Δ (%)			−2.0			+6.0			+0.0	+1.7	+12.2			+7.5			+4.5			+37.6	+12.3	+6.2
KeyeVL	Caption	41.5	13.5	20.4	26.3	10.2	14.7	46.9	14.9	22.7	19.2	9.9	43.7	18.5	26.0	37.6	15.0	21.5	55.8	19.5	28.9	25.5	16.9
	RAG	54.5	32.4	**40.6**	45.3	27.2	**34.0**	51.0	29.3	**37.2**	**37.2**	**18.1**	40.5	36.3	**38.3**	39.0	31.1	**34.6**	57.9	42.8	**49.2**	**41.0**	**24.4**
	Δ (%)			+99.0			+131.3			+63.9	+93.8	+82.8			+47.3			+60.9			+70.2	+60.8	+44.4
MiniCPM-V4.5	Caption	40.6	25.6	31.4	24.6	25.9	25.3	36.8	20.6	26.4	27.4	15.4	41.1	40.0	40.6	35.7	48.4	**41.1**	36.9	28.7	32.3	38.1	24.4
	RAG	42.4	27.1	**33.1**	26.6	29.0	**27.8**	39.8	22.5	**28.8**	**29.6**	**16.4**	42.9	41.2	**42.0**	35.3	49.0	41.0	37.2	28.9	**32.6**	**38.6**	**24.5**
	Δ (%)			+5.4			+9.9			+9.1	+8.0	+6.5			+3.4			−0.2			+0.9	+1.3	+0.4
Ovis2.5	Caption	47.8	25.5	**33.2**	61.3	30.1	**40.3**	54.0	23.9	33.1	**35.6**	**13.9**	43.4	43.4	43.4	47.5	45.2	**46.4**	63.0	50.0	55.8	48.7	**28.3**
	RAG	48.1	24.3	32.3	58.4	27.4	37.3	57.2	24.0	**33.8**	34.5	13.5	44.0	43.9	**43.9**	47.4	45.4	46.3	63.2	50.3	**56.0**	**48.9**	27.9
	Δ (%)			−2.7			−7.4			+2.1	−3.1	−2.9			+1.2			−0.2			+0.4	+0.4	−1.4
Qwen2.5-VL	Caption	41.0	29.3	34.2	53.7	35.0	42.4	67.4	40.3	**50.4**	42.7	**20.5**	49.4	49.1	49.3	48.3	42.3	45.1	53.7	42.4	47.4	47.2	21.8
	RAG	42.3	30.6	**35.5**	63.4	41.7	**50.3**	59.3	35.7	44.6	**43.7**	19.9	52.8	52.8	**52.8**	51.4	44.6	**47.8**	55.5	43.8	**49.0**	**49.8**	**23.3**
	Δ (%)			+3.8			+18.6			−11.5	+2.3	−2.9			+7.1			+6.0			+3.4	+5.5	+6.9

6 DISCUSSION, LIMITATIONS, AND CONCLUSION

We introduced WSGG and ACTIONGENOME4D to benchmark persistent spatiotemporal scene graph generation from monocular video, alongside WorldWise++, which couples persistent object queries with image-grounded entity decoding to significantly improve relationship recovery for both observed and unobserved entities. However, evaluations highlight key challenges, including reliance on monocular reconstruction quality, disparate SGDet matching protocols, and a substantial gap between unlocalized and localized entities in predictions. Future work will target unified geometric evaluation protocols, online multi-agent reasoning, and scaling to downstream embodied AI tasks.

AI USE STATEMENT

In this work, we used generative AI tools (Claude and Claude Code by Anthropic; ChatGPT and Codex by OpenAI) to clean and reformat the dataset, namely to write scripts that clean, convert, and reformat the dataset and annotation files. Separately from these tools, generative models are documented components of our research pipeline: open-source VLMs propose the relationship labels of unobserved objects, which serve as pseudo-labels in the training split and are manually corrected in the test split (Appendices E.1 and E.2.2), and the zero-shot experiments evaluate MLLMs as baselines (Section 5.2 and Appendix H). Additionally, we used generative AI tools to create and modify figures, charts, and tables, to edit author-written text for readability, and to identify and summarize related literature and format references. We have reviewed all AI-assisted work: the authors inspected the data-processing scripts and checked their outputs against the source annotations, checked every figure and table against the underlying result files, and verified each AI-suggested reference against the original publication, and the bibliography is curated manually.

REFERENCES

Elena Agliari and Giordano De Marzo. Tolerance versus synaptic noise in dense associative memories, 2020. URL https://arxiv.org/abs/2007.02849.

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, 2025. URL https://arxiv.org/abs/2511.21631.

Ermanno Bartoli, Buwei He, Dennis Rotondi, Sebastian Koch, Federico Tombari, Kai O. Arras, Patric Jensfelt, Yixi Cai, and Iolanda Leite. GESTO: Human-centric spatio-temporal memory for reasoning in dynamic scenes, 2026. URL https://arxiv.org/abs/2608.10886.

Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models, 2025. URL https://arxiv.org/abs/2406.13642.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields, 2017. URL https://arxiv.org/abs/1611.08050.

Iacopo Catalano, Julio A. Placed, Javier Civera, and Jorge Peña Queralta. Kairos: Grounded forecasting of presence and directional flow in 4d scene graphs, 2026. URL https://arxiv.org/abs/2609.27467.

Yue Chang, Rufeng Chen, Zhaofan Zhang, Yi Chen, Yifan Tian, and Sihong Xie. RAG-3DSG: Enhancing 3d scene graphs with re-shot guided retrieval-augmented generation, 2026. URL https://arxiv.org/abs/2601.10168.

Hamza Tahir Chaudhry, Jacob A. Zavatone-Veth, Dmitry Krotov, and Cengiz Pehlevan. Long sequence hopfield memory, 2023. URL https://arxiv.org/abs/2306.04532.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities, 2024. URL https://arxiv.org/abs/2401.12168.

Kaihua Chen, Deva Ramanan, and Tarasha Khurana. Using diffusion priors for video amodal segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22890–22900, 2025a. URL https://openaccess.thecvf.com/content/CVPR2025/html/Chen_Using_Diffusion_Priors_for_Video_Amodal_Segmentation_CVPR_2025_paper.html.

Pingyi Chen, Yujing Lou, Shen Cao, Jinhui Guo, Lubin Fan, Yue Wu, Lin Yang, Lizhuang Ma, and Jieping Ye. Sd-vlm: Spatial measuring and understanding with depth-encoded vision-language models, 2025b. URL https://arxiv.org/abs/2509.17664.

Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation, 2019. URL https://arxiv.org/abs/1903.03326.

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, 2024. URL https://arxiv.org/abs/2406.01584.

David G. Clark. Transient dynamics of associative memory models, 2025. URL https://arxiv.org/abs/2506.05303.

Yuren Cong, Wentong Liao, Hanno Ackermann, Bodo Rosenhahn, and Michael Ying Yang. Spatial-temporal transformer for dynamic scene graph generation, 2021. URL https://arxiv.org/abs/2107.12309.

Yuren Cong, Jinhui Yi, Bodo Rosenhahn, and Michael Ying Yang. Ssgvs: Semantic scene graph-to-video synthesis, 2022. URL https://arxiv.org/abs/2211.06119.

Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory D. Hager, Federico Tombari, and Christian Rupprecht. Semantic image manipulation using scene graphs, 2020. URL https://arxiv.org/abs/2004.03677.

Kaize Ding, Jianling Wang, Jundong Li, Kai Shu, Chenghao Liu, and Huan Liu. Graph prototypical networks for few-shot learning on attributed networks. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, pp. 295–304, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450368599. doi: 10.1145/3340531.3411922. URL https://doi.org/10.1145/3340531.3411922.

Aradhya Dixit and Tianxi Liang. Semantic event graphs for long-form video question answering, 2026. URL https://arxiv.org/abs/2601.06097.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. URL https://arxiv.org/abs/2010.11929.

Yue Fan, Xiaojian Ma, Rongpeng Su, Jun Guo, Rujie Wu, Xi Chen, and Qing Li. Embodied videoagent: Persistent memory from egocentric videos and embodied sensors enables dynamic scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6342–6352, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Fan_Embodied_VideoAgent_Persistent_Memory_from_Egocentric_Videos_and_Embodied_Sensors_ICCV_2025_paper.html.

Azade Farshad, Sabrina Musatian, Helisa Dhamo, and Nassir Navab. Migs: Meta image generation from scene graphs, 2021. URL https://arxiv.org/abs/2110.11918.

Changqun Feng, Wangxiandi Yin, Xin Hu, Lei Zhao, Dongyang Zhang, and Tao He. GEODE: Geometry-guided discrete diffusion for open-vocabulary 3d scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 7143–7153, 2026. URL https://openaccess.thecvf.com/content/CVPR2026F/html/Feng_GEODE_Geometry-Guided_Discrete_Diffusion_for_Open-Vocabulary_3D_Scene_Graph_Generation_CVPRF_2026_paper.html.

Mingtao Feng, Haoran Hou, Liang Zhang, Ziiie Wu, Yulan Guo, and Ajmal Mian. 3d spatial multimodal knowledge accumulation for scene graph prediction in point cloud. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9182–9191, 2023. doi: 10.1109/CVPR52729.2023.00886.

Shengyu Feng, Subarna Tripathi, Hesham Mostafa, Marcel Nassar, and Somdeb Majumdar. Exploiting long-term dependencies for generating dynamic scene graphs, 2022. URL https://arxiv.org/abs/2112.09828.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL https://doi.org/10.1145/358669.358692.

Andreas Fürst, Elisabeth Rumetshofer, Johannes Lehner, Viet Tran, Fei Tang, Hubert Ramsauer, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto-Nemling, and Sepp Hochreiter. Cloob: Modern hopfield networks with infoloob outperform clip, 2022. URL https://arxiv.org/abs/2110.11316.

Ziqi Gao, Jieyu Zhang, Wisdom Oluchi Ikezogwo, Jae Sung Park, Tario G. You, Daniel Ogbu, Chenhao Zheng, Weikai Huang, Yinuo Yang, Winson Han, Quan Kong, Rajat Saini, and Ranjay Krishna. Synthetic Visual Genome 2: Extracting large-scale spatio-temporal scene graphs from videos, 2026. URL https://arxiv.org/abs/2602.23543.

Luzhou Ge, Xiangyu Zhu, Jinyan Liu, and Xuesong Li. DGSG-Mind: Dynamic 3d gaussian scene graphs for long-term scene understanding and grounding, 2026. URL https://arxiv.org/abs/2605.29879.

Shijie Geng, Peng Gao, Moitreya Chatterjee, Chiori Hori, Jonathan Le Roux, Yongfeng Zhang, Hongsheng Li, and Anoop Cherian. Dynamic graph representation learning for video dialog via multi-modal shuffled transformers, 2021. URL https://arxiv.org/abs/2007.03848.

Nicolas Gorlo, Lukas Schmid, and Luca Carlone. Describe anything anywhere at any moment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 35002–35013, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Gorlo_Describe_Anything_Anywhere_At_Any_Moment_CVPR_2026_paper.html.

Zexue He, Leonid Karlinsky, Donghyun Kim, Julian McAuley, Dmitry Krotov, and Rogerio Feris. Camelot: Towards large language models with training-free consolidated associative memory, 2024. URL https://arxiv.org/abs/2402.13449.

John Helsby, Yi Yang, Bodo Rosenhahn, and Michael Ying Yang. OvDSGG: End-to-end open-vocabulary dynamic scene graph generation, 2026. URL https://arxiv.org/abs/2608.14835.

Benjamin Hoover, Duen Horng Chau, Hendrik Strobelt, Parikshit Ram, and Dmitry Krotov. Dense associative memory through the lens of random features, 2024a. URL https://arxiv.org/abs/2410.24153.

Benjamin Hoover, Hendrik Strobelt, Dmitry Krotov, Judy Hoffman, Zsolt Kira, and Duen Horng Chau. Memory in plain sight: Surveying the uncanny resemblances of associative memories and diffusion models, 2024b. URL https://arxiv.org/abs/2309.16750.

Hao-Yu Hou, Chun-Yi Lee, Motoharu Sonogashira, and Yasutomo Kawanishi. Fross: Faster-than-real-time online 3d semantic scene graph generation from rgb-d images, 2025a. URL https://arxiv.org/abs/2507.19993.

Jiawei Hou, Xiangyang Xue, and Taiping Zeng. Hi-Dyna Graph: Hierarchical dynamic scene graph for robotic autonomy in human-centric environments, 2025b. URL https://arxiv.org/abs/2506.00083.

Jerry Yao-Chieh Hu, Dennis Wu, and Han Liu. Provably optimal memory capacity for modern hopfield models: Transformer-compatible dense associative memories as spherical codes, 2024. URL https://arxiv.org/abs/2410.23126.

Jiani Huang, Amish Sethi, Matthew Kuo, Mayank Keoliya, Neelay Velingker, JungHo Jung, Ser-Nam Lim, Ziyang Li, and Mayur Naik. Esca: Contextualizing embodied agents via scene-graph generation, 2025a. URL https://arxiv.org/abs/2510.15963.

Zeyi Huang, Yuyang Ji, Xiaofang Wang, Nikhil Mehta, Tong Xiao, Donghyun Lee, Sigmund Vanvalkenburgh, Shengxin Zha, Bolin Lai, Licheng Yu, Ning Zhang, Yong Jae Lee, and Miao Liu. Building a mind palace: Structuring environment-grounded semantic graphs for effective long video analysis with LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24169–24179, 2025b. URL https://openaccess.thecvf.com/content/CVPR2025/html/Huang_Building_a_Mind_Palace_Structuring_Environment-Grounded_Semantic_Graphs_for_Effective_CVPR_2025_paper.html.

Jinbae Im, JeongYeon Nam, Nokyung Park, Hyungmin Lee, and Seunghyun Park. Egtr: Extracting graph from transformer for scene graph generation, 2024. URL https://arxiv.org/abs/2404.02072.

Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as composition of spatio-temporal scene graphs, 2019. URL https://arxiv.org/abs/1912.06992.

Minseok Kang, Minhyeok Lee, Minjung Kim, Jungho Lee, Donghyeong Kim, Sungmin Woo, Inseok Jeon, and Sangyoun Lee. Revisiting weakly-supervised video scene graph generation via pair affinity learning, 2026. URL https://arxiv.org/abs/2603.21559.

Christina Kassab, Hyeonjae Gil, Matías Mattamala, Ayoung Kim, and Maurice Fallon. LEXI-SG: Monocular 3d scene graph mapping with room-guided feed-forward reconstruction, 2026. URL https://arxiv.org/abs/2605.13741.

Kibum Kim, Kanghoon Yoon, Jaehyeong Jeon, Yeonjun In, Jinyoung Moon, Donghyun Kim, and Chanyoung Park. Llm4sgg: Large language models for weakly supervised scene graph generation, 2024. URL https://arxiv.org/abs/2310.10404.

Kibum Kim, Kanghoon Yoon, Yeonjun In, Jaehyeong Jeon, Jinyoung Moon, Donghyun Kim, and Chanyoung Park. Weakly supervised video scene graph generation via natural language supervision, 2025. URL https://arxiv.org/abs/2502.15370.

Seok-Young Kim, Abdelrahman Elskhawy, Taewook Ha, Dooyoung Kim, Eunjae Shin, Benjamin Busam, and Woontack Woo. DeWorldSG: Depth-aware 3d semantic scene graph generation via world-model priors, 2026. URL https://arxiv.org/abs/2607.00889.

Ue-Hwan Kim, Jin-Man Park, Taek-jin Song, and Jong-Hwan Kim. 3-d scene graph: A sparse and semantic representation of physical environments for intelligent agents. *IEEE Transactions on Cybernetics*, 50(12):4921–4933, December 2020. ISSN 2168-2275. doi: 10.1109/tcyb.2019.2931042. URL http://dx.doi.org/10.1109/TCYB.2019.2931042.

Yirum Kim and Ue-Hwan Kim. OP3DSG: Open-vocabulary part-aware 3d scene graph generation for real-world environments, 2026. URL https://arxiv.org/abs/2606.29786.

Sebastian Koch, Pedro Hermosilla, Narunas Vaskevicius, Mirco Colosi, and Timo Ropinski. Sgrec3d: Self-supervised 3d scene graph learning via object-level scene reconstruction, 2023. URL https://arxiv.org/abs/2309.15702.

Leo Kozachkov, Jean-Jacques Slotine, and Dmitry Krotov. Neuron–astrocyte associative memory. *Proceedings of the National Academy of Sciences*, 122(21):e2417788122, 2025. doi: 10.1073/pnas.2417788122. URL https://www.pnas.org/doi/abs/10.1073/pnas.2417788122.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Fei-Fei Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations, 2016. URL https://arxiv.org/abs/1602.07332.

Andrey Kurenkov, Michael Lingelbach, Tanmay Agarwal, Emily Jin, Chengshu Li, Ruohan Zhang, Li Fei-Fei, Jiajun Wu, Silvio Savarese, and Roberto Martín-Martín. Modeling dynamic environments with scene graph memory, 2023. URL https://arxiv.org/abs/2305.17537.

Huy Le, Nhat Chung, Tung Kieu, Jingkang Yang, and Ngan Le. UNO: Unifying one-stage video scene graph generation via object-centric visual representation learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2797–2807, 2026. URL https://openaccess.thecvf.com/content/WACV2026/html/Le_UNO_Unifying_One-stage_Video_Scene_Graph_Generation_via_Object-Centric_Visual_WACV_2026_paper.html.

Phillip Y. Lee, Jihyeon Je, Chanho Park, Mikaela Angelina Uy, Leonidas Guibas, and Minhyuk Sung. Perspective-aware reasoning in vision-language models via mental imagery simulation, 2025. URL https://arxiv.org/abs/2504.17207.

Chenxing Li, Yiping Duan, and Xiaoming Tao. Frequency-guided multi-level reasoning for scene graph generation in video, 2026. URL https://arxiv.org/abs/2604.17298.

Jing Li and Duygu Sarikaya. STSG-VQA: Evidence-grounded temporal question answering from surgical spatio-temporal scene graphs, 2026. URL https://arxiv.org/abs/2609.08543.

Joshua Li, Fernando Jose Pena Cantu, Emily Yu, Alexander Wong, Yuchen Cui, and Yuhao Chen. SAMJAM: Zero-shot video scene graph generation for egocentric kitchen videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 467–473, 2025a. URL https://openaccess.thecvf.com/content/CVPR2025W/MTF/html/Li_SAMJAM_Zero-Shot_Video_Scene_Graph_Generation_for_Egocentric_Kitchen_Videos_CVPRW_2025_paper.html.

Lin Li, Jun Xiao, Guikun Chen, Jian Shao, Yueting Zhuang, and Long Chen. Zero-shot visual relation detection via composite visual cues from large language models, 2023. URL https://arxiv.org/abs/2305.12476.

Pengteng Li, Pinhao Song, Wuyang Li, Weiyu Guo, Huizai Yao, Yijie Xu, Dugang Liu, and Hui Xiong. See&trek: Training-free spatial prompting for multimodal large language model, 2025b. URL https://arxiv.org/abs/2509.16087.

Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer, 2022a. URL https://arxiv.org/abs/2112.12970.

Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection, 2022b. URL https://arxiv.org/abs/2203.16527.

Yanjun Li, Zhaoyang Li, Honghui Chen, and Lizhi Xu. Unbiased video scene graph generation via visual and semantic dual debiasing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19047–19056, 2025c. URL https://openaccess.thecvf.com/content/CVPR2025/html/Li_Unbiased_Video_Scene_Graph_Generation_via_Visual_and_Semantic_Dual_CVPR_2025_paper.html.

Yun Li, Yiming Zhang, Tao Lin, Xiangrui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. Sti-bench: Are mllms ready for precise spatial-temporal world understanding?, 2025d. URL https://arxiv.org/abs/2503.23765.

Ming Liao, Chao Ye, Jianing Fei, and Weiyang Lin. DSG: Dynamic 3d scene graph construction for embodied agents in changing indoor environments, 2026. URL https://arxiv.org/abs/2609.00619.

Haozhe Lin, Zequn Chen, Jinzhi Zhang, Bing Bai, Yu Wang, Ruqi Huang, and Lu Fang. Realgraph: A multiview dataset for 4d real-world context graph generation. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3735–3745, 2023. doi: 10.1109/ICCV51070.2023.00348.

Xin Lin, Chong Shi, Yibing Zhan, Zuopeng Yang, Yaqi Wu, and Dacheng Tao. TD^2-Net: Toward denoising and debiasing for dynamic scene graph generation, 2024. URL https://arxiv.org/abs/2401.12479.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection, 2024. URL https://arxiv.org/abs/2303.05499.

Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

David Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 11 2004. doi: 10.1023/B:VISI.0000029664.99615.94.

Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors, 2016. URL https://arxiv.org/abs/1608.00187.

Carlo Lucibello and Marc Mézard. Exponential capacity of dense associative memories. *Physical Review Letters*, 132(7), February 2024. ISSN 1079-7114. doi: 10.1103/physrevlett.132.077301. URL http://dx.doi.org/10.1103/PhysRevLett.132.077301.

Ke Ma, Yamin Mao, Weiming Li, Shuai Tan, Yijie Zhong, Hao Chen, Haofen Wang, and Meng Wang. R4DSG: Relative 4d scene graph memory for object-centric question answering in long egocentric video, 2026. URL https://arxiv.org/abs/2608.11017.

Wufei Ma, Yu-Cheng Chou, Qihao Liu, Xingrui Wang, Celso de Melo, Jianwen Xie, and Alan Yuille. Spatialreasoner: Towards explicit and generalizable 3d spatial reasoning, 2025a. URL https://arxiv.org/abs/2504.20024.

Wufei Ma, Luoxin Ye, Celso M de Melo, Jieneng Chen, and Alan Yuille. Spatialllm: A compound 3d-informed design towards spatially-intelligent large multimodal models, 2025b. URL https://arxiv.org/abs/2505.00788.

Yanni Ma, Hao Liu, Yun Pei, and Yulan Guo. Heterogeneous graph learning for scene graph prediction in 3d point clouds. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 274–291, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-73347-5.

Vivek Madhavaram, Vartika Sengar, Arkadipta De, and Charu Sharma. VIZOR: Viewpoint-invariant zero-shot scene graph generation for 3d scene reasoning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 8584–8595, 2026. URL https://openaccess.thecvf.com/content/WACV2026/html/Madhavaram_VIZOR_Viewpoint-Invariant_Zero-Shot_Scene_Graph_Generation_for_3D_Scene_Reasoning_WACV_2026_paper.html.

Vladislav Makarov, Mark Gizetdinov, and Dmitry Yudin. SceneGraphVLM: Dynamic scene graph generation from video with vision-language models, 2026. URL https://arxiv.org/abs/2605.13667.

Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. Spatiallm: Training large language models for structured indoor modeling, 2025. URL https://arxiv.org/abs/2506.07491.

Kirill Mazur, Marwan Taher, and Andrew J. Davison. 4d primitive-mache: Glueing primitives for persistent 4d scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7372–7381, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Mazur_4D_Primitive-Mache_Glueing_Primitives_for_Persistent_4D_Scene_Reconstruction_CVPR_2026_paper.html.

Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=37nvvqkCo5.

Xingyu Miao, Haoran Duan, Quanhao Qian, Jiuniu Wang, Yang Long, Ling Shao, Deli Zhao, Ran Xu, and Gongjie Zhang. Towards scalable spatial intelligence via 2d-to-3d data lifting, 2025. URL https://arxiv.org/abs/2507.18678.

Toki Migimatsu and Jeannette Bohg. Grounding predicates through actions, 2022. URL https://arxiv.org/abs/2109.14718.

Yukuan Min, Aming Wu, and Cheng Deng. Environment-invariant curriculum relation learning for fine-grained scene graph generation, 2023. URL https://arxiv.org/abs/2308.03282.

Raul Mur-Artal, J. M. M. Montiel, and Juan D. Tardos. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, October 2015. ISSN 1941-0468. doi: 10.1109/tro.2015.2463671. URL http://dx.doi.org/10.1109/TRO.2015.2463671.

Vinoth Kumar Muthuraj, Soumyadeep Banik, Kushal Sharma, and Hardik Jain. ProClosure: Hierarchical room-object assignment using progressive boundary closure from monocular video, 2026. URL https://arxiv.org/abs/2609.12614.

Sayak Nag, Kyle Min, Subarna Tripathi, and Amit K. Roy Chowdhury. Unbiased scene graph generation in videos, 2023. URL https://arxiv.org/abs/2304.00733.

Trong-Thuan Nguyen, Pha Nguyen, and Khoa Luu. Hig: Hierarchical interlacement graph approach to scene graph generation in video understanding, 2024. URL https://arxiv.org/abs/2312.03050.

Trong-Thuan Nguyen, Pha Nguyen, Jackson Cothren, Alper Yilmaz, and Khoa Luu. Hyperglm: Hypergraph for video scene graph generation and anticipation, 2025a. URL https://arxiv.org/abs/2411.18042.

Trong-Thuan Nguyen, Pha Nguyen, Jackson Cothren, Alper Yilmaz, Minh-Triet Tran, and Khoa Luu. THYME: Temporal hierarchical-cyclic interactivity modeling for video scene graphs in aerial footage, 2025b. URL https://arxiv.org/abs/2507.09200.

Jon Nyffeler, Federico Tombari, and Daniel Barath. Hierarchical 3d scene graphs construction outdoors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26817–26826, October 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. URL https://arxiv.org/abs/2304.07193.

Rohith Peddi, Saksham Singh, Saurabh, Parag Singla, and Vibhav Gogate. Towards scene graph anticipation, 2024. URL https://arxiv.org/abs/2403.04899.

Rohith Peddi, Saurabh, Ayush Abhay Shrivastava, Parag Singla, and Vibhav Gogate. Towards unbiased and robust spatio-temporal scene graph generation and anticipation, 2025. URL https://arxiv.org/abs/2411.13059.

Francesca Pistilli, Simone Alberto Peirone, and Giuseppe Averta. Learning to evolve scenes: Reasoning about human activities with scene graphs, 2026. URL https://arxiv.org/abs/2607.02425.

Tao Pu, Tianshui Chen, Hefeng Wu, Yongyi Lu, and Liang Lin. Spatial-temporal knowledge-embedded transformer for video scene graph generation, 2023. URL https://arxiv.org/abs/2309.13237.

Haiyi Qiu, Minghe Gao, Long Qian, Kaihang Pan, Qifan Yu, Juncheng Li, Wenjie Wang, Siliang Tang, Yueting Zhuang, and Tat-Seng Chua. STEP: Enhancing video-LLMs' compositional reasoning by spatio-temporal graph-guided self-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3284–3294, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Qiu_STEP_Enhancing_Video-LLMs_Compositional_Reasoning_by_Spatio-Temporal_Graph-guided_Self-Training_CVPR_2025_paper.html.

Adrian Ramlal and John S. Zelek. PersistGS: Differentiable physics for object permanence in 4d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4687–4696, 2026. URL https://openaccess.thecvf.com/content/CVPR2026W/GenRecon3d/html/Ramlal_PersistGS_Differentiable_Physics_for_Object_Permanence_in_4D_Gaussian_Splatting_CVPRW_2026_paper.html.

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, 2024. URL https://arxiv.org/abs/2408.00714.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks, 2016. URL https://arxiv.org/abs/1506.01497.

Ivan Rodin, Antonino Furnari, Kyle Min, Subarna Tripathi, and Giovanni Maria Farinella. Action scene graphs for long-form understanding of egocentric videos, 2023. URL https://arxiv.org/abs/2312.03391.

Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4104–4113, 2016. URL https://api.semanticscholar.org/CorpusID:1728538.

Xindi Shang, Tongwei Ren, Jingfan Guo, Hanwang Zhang, and Tat-Seng Chua. Video visual relation detection. In *Proceedings of the 25th ACM International Conference on Multimedia*, MM '17, pp. 1300–1308, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450349062. doi: 10.1145/3123266.3123380. URL https://doi.org/10.1145/3123266.3123380.

Yifan Shen, Yuanzhe Liu, Jingyuan Zhu, Xu Cao, Xiaofeng Zhang, Yixiao He, Wenming Ye, James Matthew Rehg, and Ismini Lourentzou. Fine-grained preference optimization improves spatial reasoning in vlms, 2026. URL https://arxiv.org/abs/2506.21656.

Suprosanna Shit, Rajat Koner, Bastian Wittmann, Johannes Paetzold, Ivan Ezhov, Hongwei Li, Jiazhen Pan, Sahand Sharifzadeh, Georgios Kaissis, Volker Tresp, and Bjoern Menze. Relationformer: A unified framework for image-to-graph generation, 2022. URL https://arxiv.org/abs/2203.10202.

Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding, 2016. URL https://arxiv.org/abs/1604.01753.

Evgeny Smirnov, Nikita Garaev, Vasiliy Galyuk, and Evgeny Lukyanets. Prototype memory for large-scale face representation learning. *IEEE Access*, 10:12031–12046, 2022. ISSN 2169-3536. doi: 10.1109/access.2022.3146059. URL http://dx.doi.org/10.1109/ACCESS.2022.3146059.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 4080–4090, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Elizabeth S. Spelke. Principles of object perception. *Cognitive Science*, 14(1):29–56, 1990. ISSN 0364-0213. doi: https://doi.org/10.1016/0364-0213(90)90025-R. URL https://www.sciencedirect.com/science/article/pii/036402139090025R.

Fan-Yun Sun, Weiyu Liu, Siyi Gu, Dylan Lim, Goutam Bhat, Federico Tombari, Manling Li, Nick Haber, and Jiajun Wu. Layoutvlm: Differentiable optimization of 3d layout via vision-language models, 2025. URL https://arxiv.org/abs/2412.02193.

Ivan E. Sutherland and Gary W. Hodgman. Reentrant polygon clipping. *Communications of the ACM*, 17(1):32–42, 1974. doi: 10.1145/360767.360802.

Agnese Taluzzi, Riccardo Santambrogio, Simone Mentasti, Chiara Plizzari, and Matteo Matteucci. Graph it first! enabling reasoning on long-form egocentric videos through scene graphs, 2026. URL https://arxiv.org/abs/2606.25842.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. URL https://arxiv.org/abs/2302.13971.

Guan Wang, Zhimin Li, Qingchao Chen, and Yang Liu. Oed: Towards one-stage end-to-end dynamic scene graph generation, 2024a. URL https://arxiv.org/abs/2405.16925.

Linlin Wang, Xue Yang, Zhihuang Zhou, Zhenyu Zhong, Ruiyuan Zhang, and Yansheng Li. T-STAR: A large-scale benchmark for spatio-temporal panoptic scene graph generation in satellite video, 2026a. URL https://arxiv.org/abs/2607.21228.

Qi-Wei Wang, Da-Wei Zhou, Yi-Kai Zhang, De-Chuan Zhan, and Han-Jia Ye. Few-shot class-incremental learning via training-free prototype calibration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=8NAxGDdf7H.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy, 2024b. URL https://arxiv.org/abs/2312.14132.

Wenqi Wang, Reuben Tan, Pengyue Zhu, Jianwei Yang, Zhengyuan Yang, Lijuan Wang, Andrey Kolobov, Jianfeng Gao, and Boqing Gong. Site: towards spatial intelligence thorough evaluation, 2025a. URL https://arxiv.org/abs/2505.05456.

Xuejiao Wang, Bohao Zhang, Changbo Wang, and Gaoqi He. MOSA: Motion-guided semantic alignment for dynamic scene graph generation, 2026b. URL https://arxiv.org/abs/2604.19631.

Yi Wang, Zeyu Xue, Mujie Liu, Tongqin Zhang, Yan Hu, Zhou Zhao, Chenguang Yang, and Zhenyu Lu. Open-vocabulary spatio-temporal scene graph for robot perception and teleoperation planning, 2025b. URL https://arxiv.org/abs/2509.23107.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025c.

Yu Wang, Dmitry Krotov, Yuanzhe Hu, Yifan Gao, Wangchunshu Zhou, Julian McAuley, Dan Gutfreund, Rogerio Feris, and Zexue He. M+: Extending memoryllm with scalable long-term memory, 2025d. URL https://arxiv.org/abs/2502.00592.

Yufu Wang, Yu Sun, Priyanka Patel, Kostas Daniilidis, Michael J. Black, and Muhammed Kocabas. Prompthmr: Promptable human mesh recovery, 2025e. URL https://arxiv.org/abs/2504.06397.

Zirui Wang, Ruiping Liu, Yufan Chen, Junwei Zheng, Weijia Fan, Kunyu Peng, Di Wen, Jiale Wei, Jiaming Zhang, and Rainer Stiefelhagen. SGR3 model: Scene graph retrieval-reasoning model in 3d, 2026c. URL https://arxiv.org/abs/2603.04614.

Chuan Wen, Dinesh Jayaraman, and Yang Gao. Can transformers capture spatial relations between objects?, 2024. URL https://arxiv.org/abs/2403.00729.

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence, 2025a. URL https://arxiv.org/abs/2505.23747.

Rongsen Wu, Jie Xu, Hao Zheng, Zhiyuan Xu, Zixuan Li, Shixue Cheng, and Shumao Zhang. Spatio-temporal features with global–local transformer model for video scene graph generation. *Digital Communications and Networks*, 2025b. ISSN 2352-8648. doi: https://doi.org/10.1016/j.dcan.2025.04.010. URL https://www.sciencedirect.com/science/article/pii/S2352864825000434.

Shengqiong Wu, Hao Fei, and Tat-seng Chua. Universal scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14158–14168, June 2025c.

Shengqiong Wu, Hao Fei, Jingkang Yang, Xiangtai Li, Juncheng Li, Hanwang Zhang, and Tat seng Chua. Learning 4d panoptic scene graph generation from rich 2d visual scene, 2025d. URL https://arxiv.org/abs/2503.15019.

Yaxu Xie, Abdalla Arafa, Alireza Javanmardi, Christen Millerdurai, Jia Cheng Hu, Shaoxiang Wang, Alain Pagani, and Didier Stricker. ReLaGS: Relational language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23826–23836, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Xie_ReLaGS_Relational_Language_Gaussian_Splatting_CVPR_2026_paper.html.

Zhu Xu, Ting Lei, Zhimin Li, Guan Wang, Qingchao Chen, Yuxin Peng, and Yang liu. Trkt: Weakly supervised dynamic scene graph generation with temporal-enhanced relation-aware knowledge transferring, 2025. URL https://arxiv.org/abs/2508.04943.

Jingkang Yang, Wenxuan Peng, Xiangtai Li, Zujin Guo, Liangyu Chen, Bo Li, Zheng Ma, Kaiyang Zhou, Wayne Zhang, Chen Change Loy, and Ziwei Liu. Panoptic video scene graph generation, 2023. URL https://arxiv.org/abs/2311.17058.

Jingkang Yang, Jun Cen, Wenxuan Peng, Shuai Liu, Fangzhou Hong, Xiangtai Li, Kaiyang Zhou, Qifeng Chen, and Ziwei Liu. 4d panoptic scene graph generation, 2024. URL https://arxiv.org/abs/2405.10305.

Yi Yang, Myrna Castillo, Bodo Rosenhahn, and Michael Ying Yang. PUF: Plug-and-play uncertainty-aware fusion for online 3d scene graph generation, 2026. URL https://arxiv.org/abs/2607.07170.

Yuncong Yang, Jiageng Liu, Zheyuan Zhang, Siyuan Zhou, Reuben Tan, Jianwei Yang, Yilun Du, and Chuang Gan. Mindjourney: Test-time scaling with world models for spatial reasoning, 2025. URL https://arxiv.org/abs/2507.12508.

Jin Yao, Hao Gu, Xuweiyi Chen, Jiayun Wang, and Zezhou Cheng. Open vocabulary monocular 3d object detection, 2025. URL https://arxiv.org/abs/2411.16833.

Qi Xun Yeo, Yanyan Li, and Gim Hee Lee. Statistical confidence rescoring for robust 3d scene graph generation from multi-view images, 2025. URL https://arxiv.org/abs/2508.06546.

Qi Xun Yeo, Seungjun Lee, Yan Li, and Gim Hee Lee. NoPA: Non-parametric online 3d scene graph generation, 2026. URL https://arxiv.org/abs/2607.00529.

Kanghoon Yoon, Kibum Kim, Jaehyung Jeon, Yeonjun In, Donghyun Kim, and Chanyoung Park. Ra-sgg: Retrieval-augmented scene graph generation framework via multi-prototype learning, 2024. URL https://arxiv.org/abs/2412.12788.

Tatiana Zemskova and Dmitry Yudin. 3DGraphLLM: Combining semantic graphs and large language models for 3d scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8885–8895, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zemskova_3DGraphLLM_Combining_Semantic_Graphs_and_Large_Language_Models_for_3D_ICCV_2025_paper.html.

Chenyangguang Zhang, Alexandros Delitzas, Fangjinhua Wang, Ruida Zhang, Xiangyang Ji, Marc Pollefeys, and Francis Engelmann. Open-vocabulary functional 3d scene graphs for real-world indoor spaces, 2025a. URL https://arxiv.org/abs/2503.19199.

Hang Zhang, Zhuoling Li, and Jun Liu. Scenellm: Implicit language reasoning in llm for dynamic scene graph generation, 2025b. URL https://arxiv.org/abs/2412.11026.

Jia Zhao, Ziyang Cao, Huiling Wang, Xu Wang, and Yingzhou Chen. Profusion: Multimodal prototypical networks for few-shot learning with feature fusion. *Symmetry*, 17(5), 2025. ISSN 2073-8994. doi: 10.3390/sym17050796. URL https://www.mdpi.com/2073-8994/17/5/796.

Shu Zhao and Huijuan Xu. Less is more: Toward zero-shot local scene graph generation via foundation models, 2023. URL https://arxiv.org/abs/2310.01356.

Duo Zheng, Shijia Huang, Yanyang Li, and Liwei Wang. Learning from videos for 3d world: Enhancing mllms with 3d vision geometry priors, 2025. URL https://arxiv.org/abs/2505.24625.

Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. Learning to generate scene graph from natural language supervision, 2021. URL https://arxiv.org/abs/2109.02227.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5745–5753, 2019.

Zijian Zhou, Miaojing Shi, and Holger Caesar. Vlprompt: Vision-language prompting for panoptic scene graph generation, 2024. URL https://arxiv.org/abs/2311.16492.

Fangrui Zhu, Hanhui Wang, Yiming Xie, Jing Gu, Tianye Ding, Jianwei Yang, and Huaizu Jiang. Struct2d: A perception-guided framework for spatial reasoning in mllms, 2025a. URL https://arxiv.org/abs/2506.04220.

Guangming Zhu, Liang Zhang, Youliang Jiang, Yixuan Dang, Haoran Hou, Peiyi Shen, Mingtao Feng, Xia Zhao, Qiguang Miao, Syed Afaq Ali Shah, and Mohammed Bennamoun. Scene graph generation: A comprehensive survey, 2022. URL https://arxiv.org/abs/2201.00443.

Yiqi Zhu, Ziyue Wang, Can Zhang, Peng Li, and Yang Liu. Cospace: Benchmarking continuous space perception ability for vision-language models, 2025b. URL https://arxiv.org/abs/2503.14161.

Zhihao Zhu. Salient temporal encoding for dynamic scene graph generation, 2025. URL https://arxiv.org/abs/2503.14524.

APPENDIX GUIDE

The supplementary material follows the pipeline from task definition to evaluation:

- **Appendices A–B** define the notation, annotation inventory, and evaluation coverage.
- **Appendix C** explains scene reconstruction; **Appendices D–E** describe geometric and semantic annotation, each presenting the automated pipeline followed by the manual correction pipeline.
- **Appendix F** describes every supervised WSGG method in full: the adapted baselines, the monocular 3D detector, and the WorldWise model family, each with its architecture figure, losses, and training parameters.
- **Appendix G** reports the extended supervised WSGG results and the WorldWise component ablation.
- **Appendix H** describes the zero-shot U-WSGG (unlocalized) and zero-shot WSGG (localized) methods, their evaluation protocols, and their results.
- **Appendix I** provides additional related work.
- **Appendix J** discusses the limitations of the task formulation, annotations, methods, and evaluation.

CONTENTS

1 Introduction **1**

2 Related work **2**

3 Notation & Problem Description **3**

4 ActionGenome4D Dataset **4**

5 Experiments **5**
5.1 Supervised WSGG . . . 6
5.2 Zero-shot U-WSGG . . . 9

6 Discussion, Limitations, and Conclusion **9**

A Task Conventions and Notation **24**
A.1 Identity, state, and benchmark scope . . . 24
A.2 Notation reference . . . 24

B Dataset Statistics **27**
B.1 Dataset scope and annotation strategy . . . 27
B.2 Object and predicate vocabularies . . . 27
B.3 Geometric and semantic annotation counts . . . 28
B.4 Dataset Splits . . . 30

C 3D Scene Reconstruction **31**
C.1 Adaptive Frame Sampling via Feature Descriptor Matching . . . 32
C.2 Feed-Forward 3D Inference via π^3 . . . 33
C.3 Static-Dynamic Scene Decomposition . . . 34
C.4 Per-Frame Geometric Alignment via Trimmed ICP . . . 35
C.5 Mask-Aware Scene Merging . . . 36

D Geometric Annotation **39**
D.1 Automated 3D Bounding Box Pipeline . . . 39
D.2 Manual Correction Pipeline . . . 43

E Semantic Annotation **47**
E.1 Automated Relationship Annotation Pipeline . . . 47
E.2 Manual Correction Pipeline . . . 49

F Supervised WSGG Methods **51**
F.1 Shared Components . . . 51
F.2 Adapted Video Scene Graph Baselines . . . 57
F.3 WorldWise Methods . . . 62

G Supervised WSGG Evaluation and WorldWise Component Ablation **71**
G.1 WorldWise and WorldWise+: supplementary comparisons . . . 71
G.2 Supervised WSGG operating points and visibility breakdown . . . 72
G.3 Qualitative supervised WSGG scene graphs . . . 72
G.4 Adapted persistent 3D-graph fusion . . . 72
G.5 WorldWise component ablation study . . . 74
G.6 Predicate composition by visibility bucket . . . 80

H Zero-Shot WSGG and U-WSGG: Methods, Evaluation Protocols, and Results **81**
H.1 Zero-Shot U-WSGG Methods . . . 81
H.2 Zero-Shot WSGG Methods . . . 88
H.3 Zero-Shot U-WSGG Evaluation Protocol, Method Comparison and Efficiency . . . 96
H.4 Zero-Shot WSGG Experiments . . . 99
H.5 Zero-shot WSGG SGDet operating points and visibility breakdown . . . 99
H.6 Qualitative zero-shot scene graphs . . . 101
H.7 External MLLM baselines and context controls . . . 101

I Extended Context and Related Work **104**
I.1 Extended Literature on Structured Scene Understanding . . . 104
I.2 Extended Literature on Vision-Language Models for Spatial Reasoning . . . 105

I.3 Neural Memory Models 106
I.4 Recent Dynamic Scene Graphs and Persistent World State 106
I.5 Direct Comparison and Evaluation Scope 107

J Limitations **109**

A Task Conventions and Notation

A.1 Identity, state, and benchmark scope

The video-level object inventory records which objects belong to a scene; the world state records their properties at a particular timestamp. An object can retain its identity while its pose, visibility, and relationships change. The world-state denoted by $\mathcal{W}^t$ describes the time-varying state of all the objects present in the video; where the observed $\mathcal{O}^t$ and unobserved objects $\mathcal{U}^t$ partition it. Note that unobserved objects are not necessarily those that are outside the frame; they can also be present inside the frame but not visible due to occlusion or other factors. We present our models and analysis, primarily in the offline scenario where the assimption is to leverage complete video. Thus, an unobserved state can use evidence from later visible frames; this is offline completion, not a causal prediction from the video prefix.

Representations. The 2D box uses the four-coordinate image representation $[x, y, w, h]$, while the 3D OBB uses eight three-dimensional corners. ActionGenome primarily has three relationship axes label groups, namely attention, spatial and contacting. The attention axis is a single-label, three-class output describing whether the person is looking at the object. Spatial and contacting axes are multilabel; their probabilities need not sum to one. The contacting axis is a label group, not an additional predicate named "contacting." Holding and touching can coexist in a multilabel annotation, whereas a positive contact label and `not_contacting` are mutually exclusive.

A.2 Notation reference

For ease of reference, Table 4 summarizes the main mathematical symbols and architectural terms. Symbols reused with a local meaning are defined in the corresponding section.

Table 4: **Mathematical notation and architectural terms.**

Symbol / Notation	Description & Formal Definition
Task Formalization	
$V_1^T = \{I^t\}_{t=1}^T$	Video segment consisting of T monocular frames
I^t	Video frame at timestamp t
$\mathcal{W}^t$	World state at timestamp t (all objects in the scene)
$\mathcal{O}^t$	Set of observed objects visible in frame I^t
$\mathcal{U}^t$	Set of unobserved objects not visible in frame I^t
w_k^t	Object k in the world state at timestamp t
$M(t)$	Total number of objects in the world state at timestamp t
$N(t)$	Number of observed objects at timestamp t
$\text{vis}(k,t) \in \{0,1\}$	Visibility indicator: 1 iff $w_k^t \in \mathcal{O}^t$
$\mathcal{C}$	Set of all object categories
$c_k^t \in \mathcal{C}$	Category of object k at timestamp t
$\mathcal{G}^t$	Frame-level scene graph over $\mathcal{O}^t$
$\mathcal{G}_{\mathcal{W}}^t$	World scene graph over $\mathcal{W}^t$
r_{ijk}^t	Relationship triplet $(w_i^t,\, p_{ijk}^t,\, w_j^t)$
Object Representation	
$\mathbf{b}_k^t \in \mathbb{R}^{8\times 3}$	Oriented 3D bounding box (8 corners) for object k at t
$\mathbf{C}_k \in \mathbb{R}^{8\times 3}$	OBB corner coordinates for object k
$\mathbf{d}_k^t \in \mathbb{R}^4$	2D bounding box in image coordinates (observed objects only)
$\bar{\mathbf{c}}_k$	Centroid of OBB k
V_k	Volume of OBB k
C_{obj}	Number of node-head class outputs (37: 36 foreground categories including person, plus background)
$\mathbf{f}_k^t \in \mathbb{R}^{d_{\text{roi}}}$	Per-object visual features (ROI pooling)
Predicates and Relationships	

Continued on next page

Table 4 (continued from previous page)

Symbol / Notation	Description & Formal Definition
$\mathcal{P}$	Set of all predicate classes
$\mathcal{P}_{\text{att}}$	Attention predicate set ($\lvert\mathcal{P}_{\text{att}}\rvert = 3$, single-label)
$\mathcal{P}_{\text{spa}}$	Spatial predicate set ($\lvert\mathcal{P}_{\text{spa}}\rvert = 6$, multi-label)
$\mathcal{P}_{\text{con}}$	Contacting predicate set ($\lvert\mathcal{P}_{\text{con}}\rvert = 17$, multi-label)
$k = (i, j)$	Person–object pair k: person (subject) i, object j
Camera and Geometry	
$\mathbf{T}^t = \begin{bmatrix} \mathbf{R}^t & \boldsymbol{\tau}^t \\ \mathbf{0}^\top & 1 \end{bmatrix}$	Homogeneous 4×4 camera-to-world SE(3) extrinsic at timestamp t
$\mathbf{R}^t \in SO(3)$	Rotation component of the camera pose
$\boldsymbol{\tau}^t \in \mathbb{R}^3$	Translation component of the camera pose
$\mathbf{R}_{\text{rel}}, \boldsymbol{\tau}_{\text{rel}}$	Relative rotation/translation between consecutive frames
(f_x, f_y, c_x, c_y)	Camera intrinsics (focal lengths and principal point)
Architectural Components	
d_{model}	Model hidden dimension
d_{struct}	Structural token dimension
d_{rel}	Relationship token dimension
$\mathbf{g}_n^{(t)}$	Structural token for object n at frame t (from GlobalStructuralEncoder)
$\mathbf{g}_{\text{global}}$	Global structural token (max-pooled over objects)
$\mathbf{p}_i^t$	3D spatial positional encoding for object i (InterObjectTransformer)
d_{ij}	Euclidean distance between objects i and j
$\hat{\mathbf{d}}_{ij}$	Unit direction vector from object i to j
ρ_{ij}	Log-volume ratio $\log V_i - \log V_j$
$\mathbf{S}^{(t)}$	Spatial PE matrix at frame t
$\mathbf{X}^{(t)}$	Input token matrix at frame t
$\mathbf{H}^{(t)} \in \mathbb{R}^{M(t) \times d_{\text{model}}}$	Output of the InterObjectTransformer at frame t
$\overline{\mathbf{v}}^{(t)}$	Object validity mask at frame t
$\mathbf{h}_n$	Refined object representation for object n
$\hat{\mathbf{y}}_n^{\text{obj}} \in \mathbb{R}^{C_{\text{obj}}}$	Predicted object class logits
$\mathbf{r}_k^t$	Relationship representation of pair $k = (i, j)$ at frame t
$\mathbf{u}_k^t$	Union ROI features for pair $k = (i, j)$
$\mathbf{e}_{c(i)}, \mathbf{e}_{c(j)}$	CLIP text embeddings of the classes of i and j (ground truth in PredCls, predicted in SGDet)
$\mathbf{R}^{(t)}$	Refined relationship tokens after RelTransformer
$\mathbf{e}_\tau$	Temporal positional embedding at position τ
LKS Memory Buffer (W-STTran, W-STTran++, W-DsgDetr, W-DsgDetr++)	
$\mathbf{m}_n^{(t)}$	Buffered feature for object n at frame t
τ^*	Nearest visible frame for object n
$\Delta_n^{(t)}$	Staleness: $\lvert t - \tau^* \rvert$ for object n
Camera and Motion Encoders (W-STTran++, W-DsgDetr++, WorldWise)	
d_{camera}	Camera feature dimension
$\mathbf{c}_{\text{cam}}$	Global camera pose token
$\mathbf{c}_n$	Per-object camera-relative token
d_{motion}	Motion feature dimension
$\boldsymbol{\mu}_n^{(t)}$	Motion token for object n at frame t
$\mathbf{e}_t$	Ego-motion token at frame t (CameraTemporalEncoder; W-DsgDetr++, WorldWise)
Temporal Object Encoding (W-DsgDetr, W-DsgDetr++)	
d_{memory}	Temporal memory dimension
$\tilde{\mathbf{X}}_n$	Temporally refined token for object n after TemporalObjectEncoder
WorldWise (Masked Completion)	

Continued on next page

Table 4 (continued from previous page)

Symbol / Notation	Description & Formal Definition
$\mathbf{e}_{\text{[MASK]}}$	Learnable visual embedding for naturally unobserved or artificially masked objects
p_{mask}	Training-time masking fraction for visible objects
$\hat{\mathbf{x}}_n^{(t)}$	Retrieved (filled-in) token after AssociativeRetriever
$\hat{\mathbf{f}}_n^{(t)}$	Reconstructed visual features for masked object
$\mathbf{e}_{\text{vis}}[\cdot]$	Learned visibility embedding
λ_{recon}	Reconstruction loss weight
λ_{dom}	Dominance factor for reconstruction loss
Loss Functions	
$\mathcal{L}_{\text{W-STTran}}$	Total loss for the W-STTran method
$\mathcal{L}_{\text{W-STTran++}}$	Total loss for the W-STTran++ method
$\mathcal{L}_{\text{W-DsgDetr}}$	Total loss for the W-DsgDetr method
$\mathcal{L}_{\text{W-DsgDetr++}}$	Total loss for the W-DsgDetr++ method
$\mathcal{L}_{\text{WorldWise}}$	Total loss for the WorldWise method
$\mathcal{L}_r^{\text{vis}}$	Loss for predicate r on visible pairs (clean GT)
$\mathcal{L}_r^{\text{unseen}}$	Loss for predicate r on unseen pairs (pseudo-labels)
$\mathcal{L}_{\text{node}}$	Node classification loss (SGDet/SGCls only)
$\mathcal{L}_{\text{recon}}$	Feature reconstruction MSE loss (WorldWise)
$\mathcal{L}_{\text{sim}}$	Simulated-unseen relationship loss (WorldWise)
$\mathcal{L}_{\text{SG}}$	Visible-pair relationship loss in the reference WorldWise objective
λ_{vlm}	Weight for unseen-pair VLM pseudo-label loss
3D Scene Construction	
π^3	Feed-forward monocular 3D reconstruction model
$\mathcal{P}_{\text{local}}^t$	Local 3D point cloud at frame t
$\mathcal{P}_{\text{world}}^t$	World-frame 3D point cloud at frame t
$\mathcal{S}$	Static scene point cloud (inpainted background)
$\mathcal{F}^t$	Dynamic scene point cloud at frame t
Floor Alignment and Coordinate Transforms	
s_g	Global floor similarity scale factor
$\mathbf{R}_g, \boldsymbol{\tau}_g$	Global floor similarity rotation and translation
$\mathbf{R}_{\text{align}}, \boldsymbol{\tau}_{\text{align}}$	World-to-floor alignment transform
$\mathbf{R}_{\text{final}}, \boldsymbol{\tau}_{\text{final}}$	World-to-final coordinate transform
$\mathbf{T}_\delta$	Manual floor correction transform (rotation $\mathbf{R}_\delta$, translation $\boldsymbol{\tau}_\delta$, scale $\mathbf{s}_\delta$)
$\mathbf{T}_{\text{XY}}$	Automated XY-plane alignment transform
$\hat{\mathbf{n}}$	Estimated floor-plane normal
Geometric Annotation	
Grounding DINO	Zero-shot open-vocabulary 2D detector
SAM2	Segment Anything Model 2 (image and video modes)
SMPL	Parametric body mesh model (floor estimation)
Evaluation Metrics	
R@K	Recall at K
mR@K	Mean recall at K (averaged over predicate classes)
PredCls	Predicate classification (GT labels and localizations given)
SGDet	Scene graph detection (object categories and relationships predicted; localization and matching depend on the protocol)

B Dataset Statistics

B.1 Dataset scope and annotation strategy

ActionGenome Ji et al. (2019) provides frame-level 2D bounding-box annotations and human–object relationship labels for videos sourced from the Charades Sigurdsson et al. (2016). ActionGenome4D preserves the original train/test assignments for retained videos (see Table 5). The original ActionGenome videos are enriched with annotations along three axes:

1. **Geometric Annotations:** Per-frame 3D point clouds reconstructed via π^3 Wang et al. (2025c), floor-aligned via SMPL-based similarity estimation, and enclosed by oriented 3D bounding boxes (OBBs) in a shared world coordinate frame (Section D.1).
2. **Semantic Annotations:** Relationship annotations expanded from observed-only objects ($\mathcal{O}^t$) to the full world state ($\mathcal{W}^t = \mathcal{O}^t \cup \mathcal{U}^t$), covering attention, spatial, and contacting predicates for every (person, object) pair at every annotated frame (Section E.1.1).
3. **Camera Poses:** Camera-to-world SE(3) poses $\{\mathbf{T}^t\}_{t=1}^{T}$ aligned via per-frame trimmed-ICP alignment, providing ego-motion trajectories for each video (Section C).

Annotation Strategy. Annotation provenance differs by split and modality. Both splits start from automated geometric proposals; semantic labels for visible objects retain the original ActionGenome annotations, while unobserved relationships are initially generated by a VLM:

- **Pseudo-annotations (Automated Generation):** For *semantic annotations*, relationship labels for unobserved objects are generated automatically using a VLM-based pipeline (Section E.1.1). A Graph-RAG approach first retrieves relevant spatio-temporal context from a precomputed coarse event graph, which is then used to estimate pseudo-labels via a VLM-based discriminative verification step. For *geometric annotations*, per-frame 3D point clouds, camera poses, and initial world-frame 3D OBBs are automatically generated across both splits via feed-forward monocular reconstruction (π^3) and similarity-based floor alignment (Section D.1, Section C).
- **Manual Corrections (Human Verification):** For *semantic annotations*, unobserved training relationships retain VLM pseudo-labels; unobserved test relationships are reviewed and corrected with the relationship interface (Section E.2 and Section E.2.2). For *geometric annotations*, training boxes are fitted automatically, incorporating available manual floor corrections. This does not imply exhaustive manual review of individual training boxes. The test workflow refits boxes after floor correction and supports interactive box and trajectory correction (Section D.2.1, Section D.2.2). The reported statistics do not quantify individual-box review coverage by split; these workflow descriptions therefore do not establish that every box was individually reviewed.

B.2 Object and predicate vocabularies

ActionGenome4D inherits the ActionGenome predicate vocabulary, organised into three disjoint axes (label-groups): Attention is a single-label classification (exactly one label per object pair per frame), while spatial and contacting are multi-label (an object pair may simultaneously exhibit multiple spatial or contacting relationships, e.g., a person can be both `holding` and `touching` an object). Figures 5 and 6 visualize the label support, its division between observed and unobserved objects, and the co-occurrence structure.

Table 5: **Comparison.**

Property / Dimension	ActionGenome	ActionGenome4D (Ours)
Problem Formulation & Scope		
Object scope per frame	$\mathcal{O}^t$ only	$\mathcal{W}^t = \mathcal{O}^t \cup \mathcal{U}^t$
Relationship coverage	Observed person–object	Observed & unobserved pairs
Coordinate frame	2D image plane	3D world (reconstruction units)
Object localization	2D bounding boxes	2D BBox + oriented 3D OBB
Camera trajectory	✗	✓ (SE(3) per frame)
3D scene reconstruction	✗	✓ (per-frame point clouds)
Annotation Taxonomy		
Predicate axes	Attention, Spatial, Contacting	Attention, Spatial, Contacting
Attention labels	3 (single-label)	3 (single-label)
Spatial labels	6 (multi-label)	6 (multi-label)
Contacting labels	17 (multi-label)	17 (multi-label)
Object categories	35 (+ person)	35 (+ person)
Unobserved object labels	✗	✓
Dataset Scale		
Total videos	9,601	9,027
Train / Test split	7,787 / 1,814	7,516 / 1,511
Annotated video frames	288,782	224,585

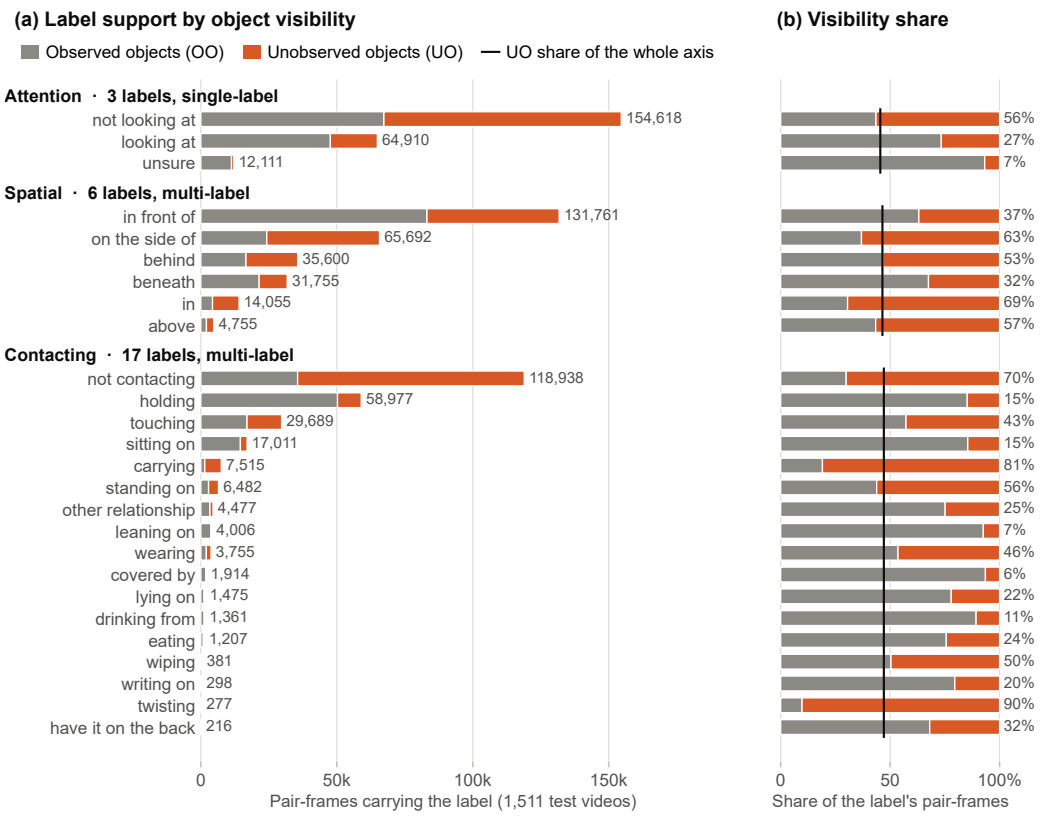


Figure 5: **Predicate support and visibility on the test split.** (a) Pair-frames carrying each label, divided into observed (OO) and unobserved (UO) objects. (b) UO share of each label; the black rule marks the UO share over the whole axis. Every axis is long-tailed, and the UO share varies across labels.

Object Categories. The object vocabulary consists of 35 categories (excluding `person`) inherited from ActionGenome. These span common indoor objects such as furniture (`chair`, `table`, `sofa`, `bed`), personal items (`phone/camera`, `bag`, `book`, `laptop`), kitchenware (`cup/glass/bottle`, `dish`, `sandwich`), and room elements (`door`, `doorway`, `window`, `mirror`, `television`). Including `person`, the foreground category set comprises 36 classes. The SGDet node classifier has 37 outputs: these 36 foreground classes and one background class (index 0).

Predicate Distribution. The predicate label distributions in ACTIONGENOME4D exhibit long-tail characteristics consistent with real-world relationship frequencies. For the **attention** axis, `not_looking_at` dominates (especially for unobserved objects, where the person is typically not attending to off-screen objects), followed by `looking_at` and `unsure`. For the **contacting** axis, `not_contacting` is the most frequent label (again driven by unobserved objects with no physical contact), while labels such as `holding`, `touching`, and `sitting_on` cover the active interactions. For the **spatial** axis, `in_front_of` and `on_the_side_of` are common due to typical indoor scene layouts, while `beneath` and `in` are relatively rare.

B.3 GEOMETRIC AND SEMANTIC ANNOTATION COUNTS

The geometric annotation pipeline (Section D.2.2) produces world-frame OBBs for annotated objects. Key statistics of the geometric annotations are:

- **Total 3D OBBs:** 864,629 object OBBs and 224,574 person OBBs across both splits.

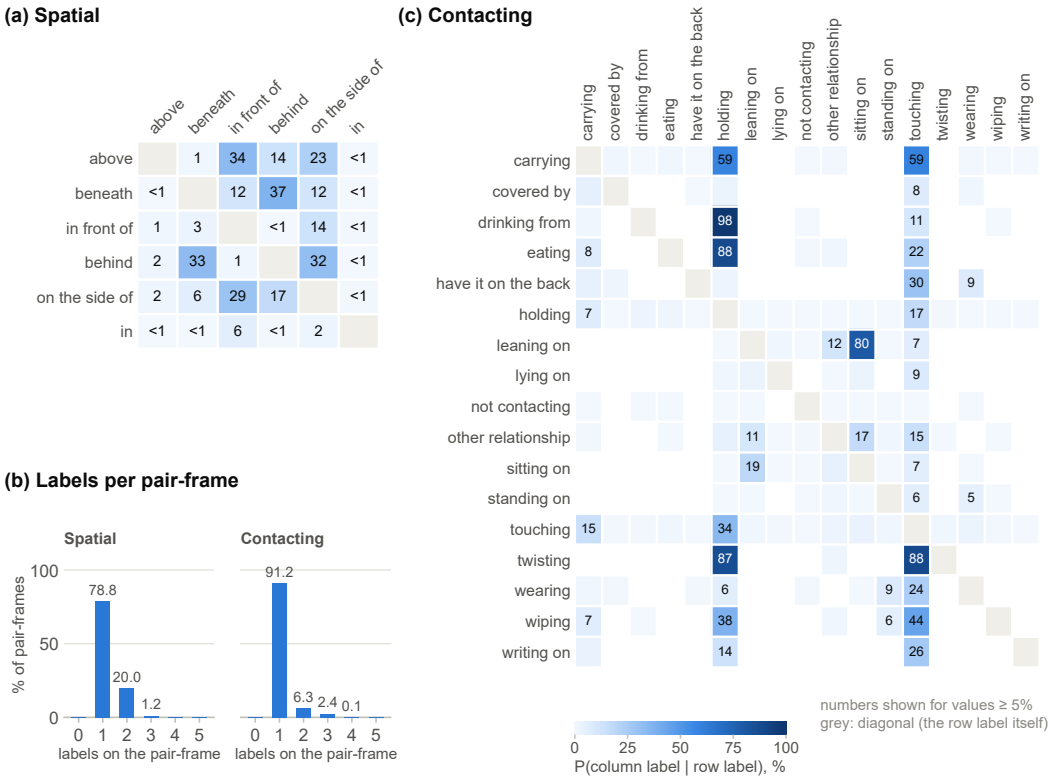


Figure 6: **Within-axis label co-occurrence on the test split.** Each cell of (a) the spatial and (c) the contacting matrix gives the percentage of pair-frames with the row label that also carry the column label, $P(\text{column} \mid \text{row})$. Blank cells never co-occur, and the grey diagonal is the row label itself. (b) Number of labels per pair-frame: 21.2% of pair-frames carry more than one spatial label and 8.8% more than one contacting label. Co-occurrences follow physically compatible combinations: `drinking_from` and `eating` co-occur with `holding` in 98% and 88% of their pair-frames, `leaning_on` with `sitting_on` in 80%, and `behind` with `beneath` and `on_the_side_of` in 33% and 32%.

- **OBB fitting:** Every OBB is floor-parallel, with a single yaw rotation about the floor normal. The automated proposal pipeline fits a minimum-area rectangle to the points of the minimum-volume candidate of a multiscale erosion sweep ($\{0, 1, \ldots, 10\}$ px kernels) and completes each object's track across the annotated frames (Section D.1.5). Training geometry incorporates available manual floor corrections. In the test workflow, boxes are refit after manual floor correction, with yaw estimated by 2D PCA on the floor plane, and each static object is represented by one axis-aligned box (Sections D.2.1 and D.2.2).
- **Coordinate system:** All OBBs are expressed in the final coordinate system (floor-aligned, centered at the floor origin, in π^3 reconstruction units with the per-video metric scale stored alongside), together with per-frame camera poses.

Semantic Annotation Statistics. ACTIONGENOME4D extends the original AG relationship annotations from observed-only objects to the full world state. The key distinction is that every (person, object) pair receives relationship labels at every annotated timestamp, regardless of objects' visibility.

- **Pair-frames:** 864,797 person–object pair-frames across both splits. Each pair-frame is one annotated object at one frame, since the person is the only subject.
- **Observed objects (OO):** 499,799 pair-frames whose object is visible ($w_k^t \in \mathcal{O}^t$).
- **Unobserved objects (UO):** 364,998 pair-frames whose object is unobserved ($w_k^t \in \mathcal{U}^t$), which are new in ACTIONGENOME4D. They make up 41.0%/45.5% for train/test set.
- **Avg. objects per frame (world state):** 3.9 non-person objects per frame, compared with 2.2 observed non-person objects per frame. Because every object forms one pair with the person, this is also the number of pair-frames per frame in $\mathcal{G}_{\mathcal{W}}^t$.

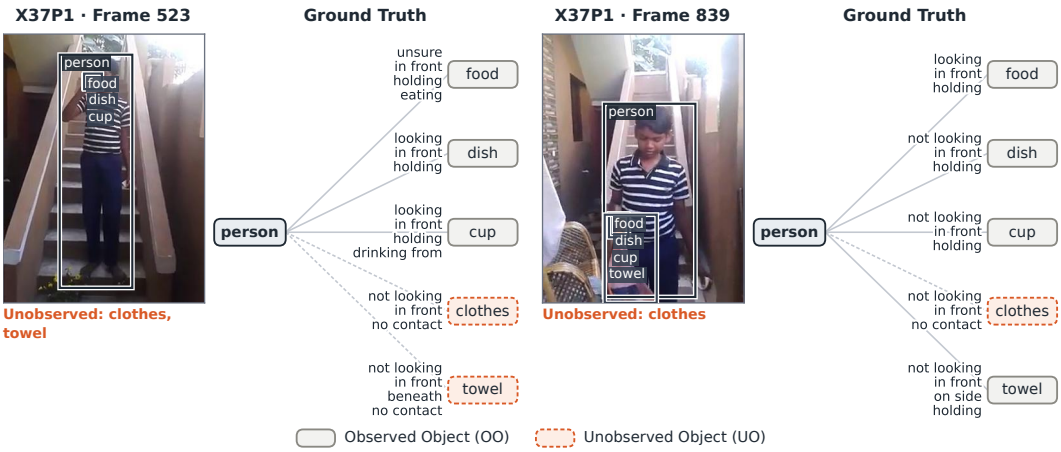


Figure 7: **Annotated world scene graphs of test video `X37P1` at frames 523 and 839.** Each graph links the person to every object of the video, observed (OO, solid grey) or unobserved (UO, dashed orange), with its attention, spatial, and contacting labels; the frames show the 2D boxes of the person and the observed objects. The towel is part of the world scene graph before it is seen: at frame 523 it is unobserved and labeled `not_looking_at`, `in_front_of`, `beneath`, and `not_contacting`, and at frame 839 it is observed and held. The clothes are unobserved in both frames. The graphs abbreviate some labels, e.g., *in front* for `in_front_of`, *on side* for `on_the_side_of`, and *no contact* for `not_contacting`.

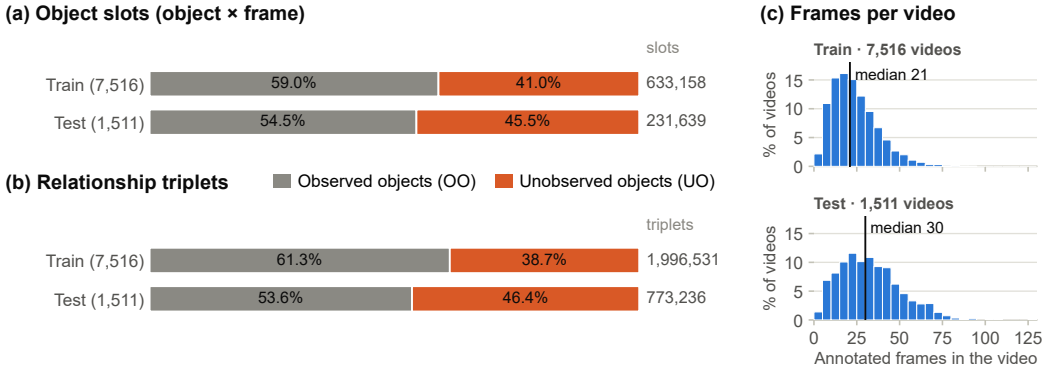


Figure 8: **Visibility composition of the ACTIONGENOME4D train and test splits.** (a) Object representations (object × frame) and (b) relationship triplets of observed (OO) and unobserved (UO) objects. Parentheses give the number of videos, and the right column gives the totals observed and unobserved object representations/triplets. Unobserved objects account for 41.0% of the training and 45.5% of the test object representations. The training split also has a smaller UO share of triplets (38.7% versus 46.4%). (c) Annotated frames per video; test videos contain more annotated frames (median 30 versus 21).

Figure 7 shows the resulting world scene graph of one test video at two frames. Every object of the video keeps its node and its relationships whether or not it is visible: the towel is annotated before it first enters the view, and the clothes throughout.

B.4 DATASET SPLITS

The train split uses VLM pseudo-annotations for unobserved object relationships; the test split uses manually corrected annotations. Figure 8 summarizes the visibility composition of each split.

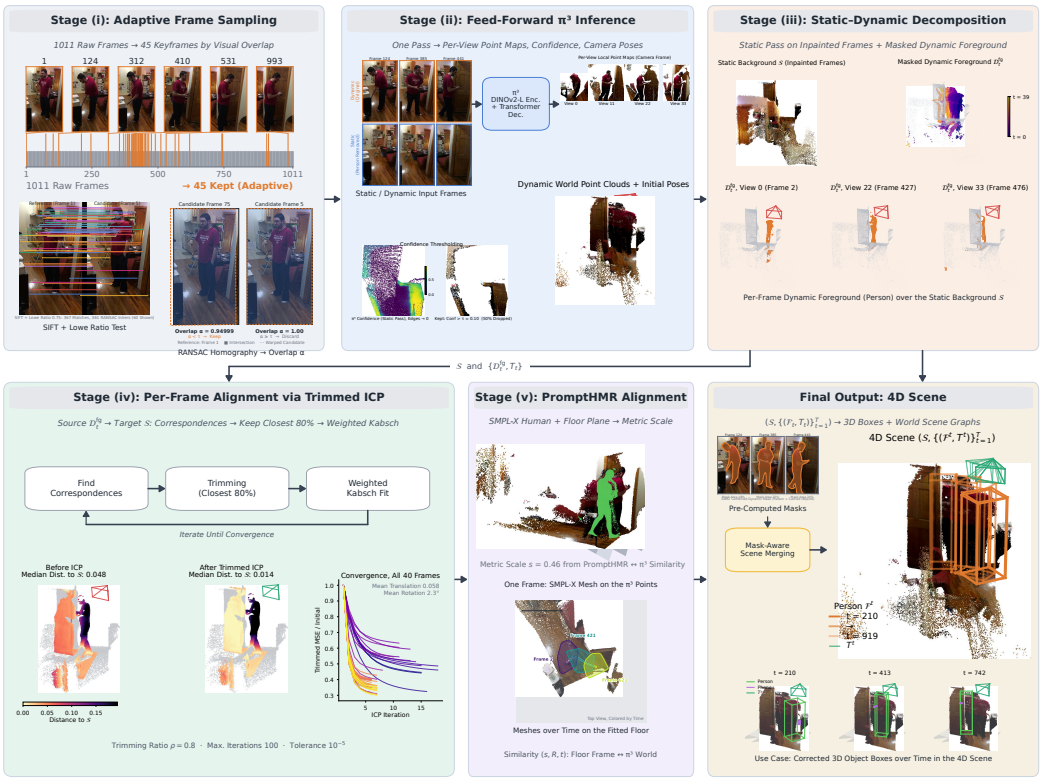


Figure 9: **4D scene construction on the ActionGenome video `0DJ6R`, shown with the actual intermediate outputs of each stage.** *Stage (i):* Adaptive sampling retains 45 of the 1,011 raw frames. SIFT matches passing Lowe's ratio test are used to estimate a homography with RANSAC. The resulting overlap α with the current reference determines whether a candidate frame is kept ($\alpha < \tau$) or discarded. *Stage (ii):* The static (inpainted) and dynamic (original) frame sets are processed by π^3, which returns per-view local point maps, per-pixel confidence (depth edges are set to zero and points below $\tau_{\text{static}} = 0.1$ are removed), and world-frame point clouds with initial camera poses $\mathbf{T}_t^{\text{dyn}}$. *Stage (iii):* The static pass yields the background $\mathcal{S}$, while the SAM2 masks isolate the per-frame dynamic foreground $\mathcal{D}_t^{\text{fg}}$ (colored by time). *Stage (iv):* Trimmed ICP registers each dynamic frame $\mathcal{D}_t$ to $\mathcal{S}$. Points are colored by their distance to $\mathcal{S}$ before and after registration, and the curves trace the trimmed MSE of every frame across iterations. *Stage (v):* The PromptHMR body mesh, aligned to the π^3 points of one frame, fixes the floor plane and the metric scale ($s = 0.46$) used downstream (Section D.1.3). The top view shows the body trajectory over all 40 keyframes on the fitted floor, with footprints at three timestamps and a 1 m scale bar. *Final output:* The pre-computed masks drive mask-aware merging into the 4D scene, shown with the person's boxes at six timestamps, and the corrected person and phone boxes at three annotated timestamps.

C 3D Scene Reconstruction

We construct 3D scenes from monocular ActionGenome Ji et al. (2019) videos using a feed-forward neural reconstruction model followed by post-hoc geometric alignment. The pipeline comprises four stages: (i) Frame Sampling and Preprocessing, (ii) Feed-forward 3D Inference via the π^3 Wang et al. (2025c) model, (iii) Static–Dynamic Scene Decomposition, and (iv) Per-frame Alignment via Trimmed Iterative Closest Point (ICP) with Weighted Kabsch Fitting. Figure 9 illustrates the full pipeline on the ActionGenome video `0DJ6R`. The panels show actual intermediate outputs, which serve as a running example throughout this section. The figure also shows, as Stage (v), the PromptHMR human and floor alignment that supplies the metric floor frame, which we describe in Section D.1.3.

4D Scene Representation. The output of the pipeline is a single 4D scene per video:

$$\text{4D Scene} = \Big(\underbrace{\mathcal{S}}_{\text{static background}} , \; \big\{ \underbrace{(\mathcal{F}^t, \mathbf{T}^t)}_{\text{per-frame dynamic}} \big\}_{t=1}^{T} \Big) \tag{5}$$

Algorithm 1 Greedy Frame Selection by Visual Overlap

Require: Frames $\{I_0, \ldots, I_{T-1}\}$, threshold τ
Ensure: Selected frame indices $\mathcal{K}$
$\mathcal{K} \leftarrow \{0\}$; $I_{\text{ref}} \leftarrow I_0$
for $t = 1, \ldots, T-1$ **do**
  $\alpha \leftarrow \text{HomographyOverlap}(I_{\text{ref}}, I_t)$
  if $\alpha < \tau$ **then** ▷ Sufficient new content
    $\mathcal{K} \leftarrow \mathcal{K} \cup \{t\}$; $I_{\text{ref}} \leftarrow I_t$
  end if
end for
return $\mathcal{K}$

where $\mathcal{S}$ is the static background point cloud (built once and shared across all frames), $\mathcal{F}^t$ is the per-frame dynamic point cloud for frame I^t, and $\mathbf{T}^t \in \mathbb{R}^{4\times4}$ is the ICP-refined camera-to-world pose. This representation supplies the geometry used to construct the world state $\mathcal{W}^t$: object detections within each frame's point cloud $\mathcal{F}^t$ yield the 3D oriented bounding boxes $\mathbf{b}_k^t$ for all interacting objects, while the refined poses $\mathbf{T}^t$ enable camera-relative reasoning for visibility determination.

C.1 Adaptive Frame Sampling via Feature Descriptor Matching

Raw ActionGenome Ji et al. (2019) videos are captured at 24–30 fps, producing substantial temporal redundancy. Rather than uniform subsampling (e.g., every k-th frame), we employ a *Feature Descriptor Sampler* that adapts its density to camera/object motion: it retains more frames during rapid motion and fewer during static shots, using SIFT feature matching Lowe (2004) and RANSAC homography estimation Fischler & Bolles (1981) to quantify visual overlap.

Homography-Based Overlap Estimation. The overlap between a retained reference frame I_A and a candidate frame I_B is estimated in four stages:

1. **SIFT extraction**: Both frames are converted to grayscale and SIFT Lowe (2004) keypoints $\mathcal{K}$ and 128-d descriptors $\mathbf{D}$ are extracted. If either frame yields < 4 keypoints, the overlap defaults to 0.
2. **Feature matching**: Descriptors in I_A are matched to I_B via brute-force k-NN (k=2) with Lowe's ratio test Lowe (2004) ($\rho = 0.75$), retaining only matches where the best candidate is substantially better than the second-best:
$$\mathcal{G} = \left\{(i,j) : \|\mathbf{d}_i^A - \mathbf{d}_{j_1}^B\|_2 \,/\, \|\mathbf{d}_i^A - \mathbf{d}_{j_2}^B\|_2 < \rho\right\}. \tag{6}$$
3. **RANSAC homography**: A projective transformation $\mathbf{H} \in \mathbb{R}^{3\times3}$ mapping I_B into I_A's coordinate frame is estimated from the good matches via RANSAC Fischler & Bolles (1981) (reprojection threshold 4.0 px, 2000 iterations, 0.995 confidence). The homography is rejected if < 4 inliers remain.
4. **Polygon intersection**: The four corners of I_B are warped into I_A's frame via $\mathbf{H}$, and the visual overlap is the normalized intersection area:
$$\alpha = \frac{\text{Area}(\mathcal{P}_A \cap \mathcal{P}'_B)}{\text{Area}(\mathcal{P}_A)} \in [0,1], \tag{7}$$
where $\mathcal{P}_A$ is the frame A rectangle and $\mathcal{P}'_B$ is the warped frame B quadrilateral, computed using the Shapely geometry library.

While ActionGenome4D scenes are not strictly planar, the homography provides a sufficiently accurate alignment for overlap estimation in indoor environments dominated by large planar surfaces (walls, floors, tables). The RANSAC inlier ratio also serves as an implicit quality measure; low inlier counts indicate strong parallax or dynamic content, both signals of significant visual change.

Greedy Adaptive Selection. Given a video with T frames and an overlap threshold $\tau = 0.95$, Algorithm 1 greedily builds the selected set $\mathcal{K}$. The reference I_{ref} changes only when a frame is retained: the reference stays fixed during static shots (skipping many near-duplicates) and advances

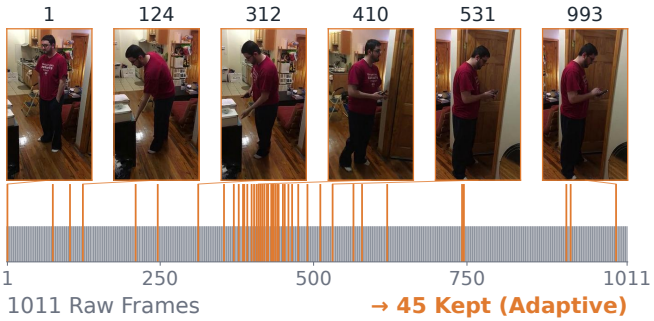


Figure 10: Adaptive sampling on 0DJ6R: 45 of the 1,011 raw frames are kept (orange).

rapidly during camera or object motion. At $\tau = 0.95$, a candidate is retained when the estimated overlap covers less than 95% of the reference image. Like keyframe selection in visual SLAM Mur-Artal et al. (2015) and SfM Schönberger & Frahm (2016), the aim is to retain informative views while avoiding near-duplicates; here the selection criterion is the explicit overlap estimate. Thus, the sampler compresses ActionGenome videos from ∼900–9000 raw frames to ∼30–150 frames. Figure 10 and Stage (i) of Figure 9 show this behavior on 0DJ6R. Frame 5 still overlaps the reference frame 1 almost entirely ($\alpha = 1.00$, 361 RANSAC inliers out of 367 ratio-test matches) and is discarded. The reference remains fixed until frame 75, the first frame whose overlap falls below the threshold ($\alpha = 0.94999 < \tau$).

Caveats: (a) To preserve all supervised training targets, every frame carrying a ground-truth bounding-box annotation is unconditionally added after overlap filtering: $\mathcal{K} \leftarrow \mathrm{sort}(\mathcal{K} \cup \mathcal{A}_{\text{annotated}})$. (b) If the resulting set contains fewer than $K_{\min} = 17$ frames, the sampler reverts to uniform subsampling at stride $\Delta = \max(1, \lfloor T/K_{\min} \rfloor)$ and re-injects annotated frames, increasing the number of input frames when overlap filtering retains too few.

C.2 Feed-Forward 3D Inference via π^3

We employ the π^3 Wang et al. (2025c) model, a feed-forward neural network for visual geometry reconstruction. Given N unposed images, π^3 jointly predicts per-pixel 3D points, per-pixel confidence scores, and camera-to-world SE(3) poses in a single forward pass, without requiring a designated reference view. We use π^3 as a black-box reconstruction module; we refer the reader to Wang et al. (2025c) for full architectural details.

Input Frame Preparation. From the selected frame set $\mathcal{K}$, we prepare two parallel sets of inputs: (a) **Dynamic frames**: The original unmodified video frames at the same indices, containing all actors and objects in their natural positions. Both sets share identical indices and resolution to enable pixel-level correspondence between the reconstructions. (b) **Static frames**: Inpainted versions in which all dynamic foreground objects (people and objects involved in interactions) have been removed via rectangular mask inpainting, depicting only the static background.

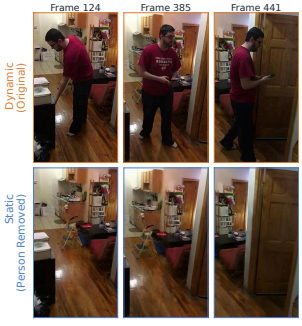


Figure 11: Dynamic (original) and static (inpainted) inputs for three retained frames of 0DJ6R.

Inference Procedure. The static and dynamic frame sets are processed in **two independent forward passes** in mixed precision (bfloat16/float16). Each pass takes N frames batched as a tensor of shape $(1, N, 3, H, W)$ and produces all outputs simultaneously. This independence is critical: the two reconstructions inhabit different, unaligned world coordinate systems, necessitating the geometric alignment described in Section C.4. For a set of N input images, the model jointly estimates the quantities summarized in Table 6.

Table 6: **Geometric representations estimated by the π^3 model** for N input views at resolution $H \times W$.

Output Quantity	Tensor Shape	Precision	Description
Local points	$(N, H, W, 3)$	float32	Per-pixel 3D coordinates in camera coordinate frame
World points	$(N, H, W, 3)$	float32	Per-pixel 3D coordinates in world coordinate frame
Confidence	$(N, H, W, 1)$	float32	Per-pixel reliability score $\in [0, 1]$
Camera poses	$(N, 4, 4)$	float32	Estimated camera-to-world SE(3) transformation matrices

Post-Processing. Two filtering steps are applied to the raw predictions before downstream use: (i) **Depth-edge suppression**: pixels at depth discontinuities (relative depth difference $> 3\%$ in a 3×3 neighborhood) have their confidence set to zero; (ii) **Confidence thresholding**: points below $\tau_{\text{static}} = 0.1$ (static background) or $\tau_{\text{frame}} = 0.01$ (per-frame dynamic) are discarded. World-frame 3D points are obtained by applying the predicted camera-to-world pose $\mathbf{T}_i$ to the local points via homogeneous transformation: $\mathbf{p}_{\text{world}} = (\mathbf{T}_i \cdot \tilde{\mathbf{p}}_{\text{local}})_{1:3}$. Figure 12 shows both filters on one static-pass view of `0DJ6R`, in which the actor's bounding rectangle covers 47% of the frame. The two filters discard 50% of the pixels, and 85% of the discarded pixels lie inside the inpainted rectangle, where 90% of the pixels fall below τ_{static}. Outside the rectangle, only 14% of the pixels are discarded. This indicates that π^3 assigns low confidence to hallucinated inpainted content, which is inconsistent across views, and that confidence thresholding removes such content before it can corrupt the static background.

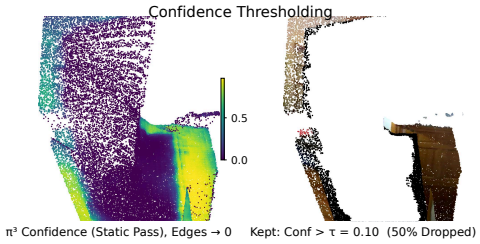


Figure 12: Static-pass confidence (edges set to zero) and the points kept at $\tau_{\text{static}} = 0.1$.

C.3 Static-Dynamic Scene Decomposition

The two-pass inference strategy produces complementary scene representations:

Static Scene. The static reconstruction uses inpainted frames as input, yielding a background point cloud from which dynamic foreground objects have been removed. Confidence and validity filters suppress unreliable reconstructed content. This point cloud is constructed by:

1. Applying a confidence threshold $\tau_{\text{static}} = 0.1$ to filter low-quality points.
2. Suppressing near-black pixels (per-channel intensity ≤ 8) that coincide with low confidence ($c < 1.0$), which removes inpainting artifacts that often manifest as dark, uncertain regions.
3. Removing all points with non-finite coordinates (NaN or Inf).

The resulting point cloud $\mathcal{S} = \{(\mathbf{s}_j, \mathbf{c}_j^{\text{rgb}})\}_{j=1}^{M}$ provides a stable geometric reference frame for subsequent alignment. The static scene is computed once per video and serves as the persistent, time-invariant component of the 4D scene. For `0DJ6R`, $\mathcal{S}$ retains $M \approx 4.9$M of the $\sim$10.2M pixels across the 40 static views (Figure 9, Stage (iii), left). Figure 13 (top) places $\mathcal{S}$ next to one of the inpainted frames it is built from, in which the actor has been removed.

Dynamic Scene. The dynamic reconstruction uses original (unmodified) frames, producing per-frame point clouds that include all actors and objects. Each frame t yields a set of world-frame 3D points $\mathcal{D}_t = \{(\mathbf{d}_k^t, \mathbf{c}_k^{t,\text{rgb}})\}$ and camera pose $\mathbf{T}_t^{\text{dyn}}$. Figure 13 and Stage (iii) of Figure 9 show the masked foreground $\mathcal{D}_t^{\text{fg}}$ of these clouds (Section C.5) for `0DJ6R`, colored by time and for three individual frames together with

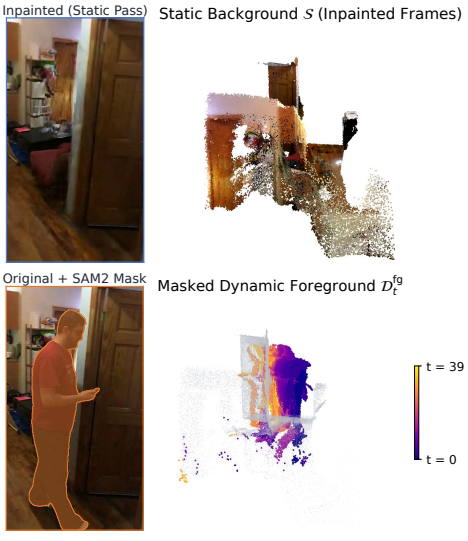


Figure 13: RGB inputs of view 22 of `0DJ6R` (left): the inpainted frame of the static pass and the original frame with its SAM2 mask. Right: the static background $\mathcal{S}$ from all inpainted frames and the masked dynamic foreground $\mathcal{D}_t^{\text{fg}}$

Algorithm 2 Trimmed ICP with Weighted Kabsch Alignment

Require: Source points $\mathbf{A} = \{\mathbf{a}_k\}_{k=1}^{K} \subset \mathbb{R}^3$ (dynamic frame)
Require: Target points $\mathbf{B} = \{\mathbf{b}_j\}_{j=1}^{M} \subset \mathbb{R}^3$ (static background)
Require: Per-point confidence weights $\{w_k\}_{k=1}^{K}$
Require: Parameters: $I_{\max} = 100$, $\epsilon = 10^{-5}$, $\rho = 0.8$
Ensure: Rigid transform $(\mathbf{R}_{\text{total}}, \boldsymbol{\tau}_{\text{total}})$

Build KD-tree $\mathcal{T}$ on target $\mathbf{B}$
$\mathbf{R}_{\text{total}} \leftarrow \mathbf{I}_3$, $\boldsymbol{\tau}_{\text{total}} \leftarrow \mathbf{0}$
$\mathbf{A}' \leftarrow \mathbf{A}$ ▷ working copy
$e_{\text{prev}} \leftarrow +\infty$
for iter $= 1$ **to** $I_{\max}$ **do**
  $(\{d_k\}, \{\text{nn}_k\}) \leftarrow \mathcal{T}.\text{query}(\mathbf{A}', k = 1)$ ▷ nearest neighbors
  $\mathcal{V} \leftarrow \{k : d_k < \infty\}$ ▷ valid correspondences
  $\delta \leftarrow \text{Percentile}(\{d_k\}_{k\in\mathcal{V}},\ \rho \cdot 100)$ ▷ trimming cutoff
  $\mathcal{V} \leftarrow \{k \in \mathcal{V} : d_k \le \delta\}$ ▷ keep closest ρ fraction
  if $|\mathcal{V}| < 10$ **then**
    break
  end if
  $(\mathbf{R}_{\text{inc}}, \boldsymbol{\tau}_{\text{inc}}) \leftarrow \text{WeightedKabsch}(\mathbf{A}'_{\mathcal{V}},\ \mathbf{B}_{\text{nn}_{\mathcal{V}}},\ \{w_k\}_{k\in\mathcal{V}})$
  $\mathbf{R}_{\text{total}} \leftarrow \mathbf{R}_{\text{inc}} \cdot \mathbf{R}_{\text{total}}$
  $\boldsymbol{\tau}_{\text{total}} \leftarrow \mathbf{R}_{\text{inc}} \cdot \boldsymbol{\tau}_{\text{total}} + \boldsymbol{\tau}_{\text{inc}}$
  $\mathbf{A}' \leftarrow (\mathbf{R}_{\text{inc}} \cdot \mathbf{A}'^{\top})^{\top} + \boldsymbol{\tau}_{\text{inc}}^{\top}$ ▷ update source
  $e \leftarrow \frac{1}{|\mathcal{V}|} \sum_{k\in\mathcal{V}} \|\mathbf{a}'_k - \mathbf{b}_{\text{nn}_k}\|^2$ ▷ MSE
  if $|e_{\text{prev}} - e| < \epsilon$ **then**
    break ▷ converged
  end if
  $e_{\text{prev}} \leftarrow e$
end for
return $(\mathbf{R}_{\text{total}}, \boldsymbol{\tau}_{\text{total}})$

their initial poses. Before alignment, the foreground of each frame is expressed in the coordinate frame of the dynamic pass rather than that of $\mathcal{S}$.

C.4 Per-Frame Geometric Alignment via Trimmed ICP

Since the static and dynamic scenes are reconstructed independently (through separate feed-forward passes of the π^3 model), their world coordinate frames are not aligned. We register each dynamic frame's point cloud to the static background via **Trimmed Iterative Closest Point (ICP)** with **confidence-weighted Kabsch fitting**. This is per-frame rigid registration: the estimated transform is applied to the points and composed with the camera pose. It is distinct from jointly optimizing camera and scene variables under a bundle-adjustment objective.

Algorithm Overview. For each frame t, the alignment proceeds according to Algorithm 2. The source cloud is the dynamic frame's point set $\mathcal{D}_t$, and the target is the static background $\mathcal{S}$.

Weighted Kabsch Algorithm. At each ICP iteration, the optimal rigid transformation aligning source correspondences $\mathbf{A}_{\mathcal{V}}$ to target correspondences $\mathbf{B}_{\mathcal{V}}$ is found by solving the weighted least-squares problem:

$$\mathbf{R}^*, \boldsymbol{\tau}^* = \underset{\mathbf{R}\in\text{SO}(3),\ \boldsymbol{\tau}\in\mathbb{R}^3}{\arg\min} \sum_{k\in\mathcal{V}} w_k \|\mathbf{R}\mathbf{a}_k + \boldsymbol{\tau} - \mathbf{b}_{\text{nn}_k}\|^2 \tag{8}$$

where w_k is the per-point confidence from the π^3 model's sigmoid-activated confidence output. The closed-form solution proceeds as follows:

1. **Weighted centroids**:

$$\boldsymbol{\mu}_A = \frac{\sum_k w_k \mathbf{a}_k}{\sum_k w_k}, \quad \boldsymbol{\mu}_B = \frac{\sum_k w_k \mathbf{b}_k}{\sum_k w_k} \tag{9}$$

2. **Cross-covariance matrix**:

$$\mathbf{H} = \sum_{k \in \mathcal{V}} w_k (\mathbf{a}_k - \boldsymbol{\mu}_A)(\mathbf{b}_k - \boldsymbol{\mu}_B)^\top \in \mathbb{R}^{3\times 3} \tag{10}$$

3. **SVD decomposition**: $\mathbf{H} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$
4. **Optimal rotation** with reflection correction:

$$\mathbf{R}^* = \mathbf{V} \cdot \mathrm{diag}\big(1,\ 1,\ \det(\mathbf{V}\mathbf{U}^\top)\big) \cdot \mathbf{U}^\top \tag{11}$$

The diagonal correction ensures $\det(\mathbf{R}^*) > 0$, i.e., $\mathbf{R}^* \in \mathrm{SO}(3)$.

5. **Optimal translation**:

$$\boldsymbol{\tau}^* = \boldsymbol{\mu}_B - \mathbf{R}^* \boldsymbol{\mu}_A \tag{12}$$

Trimming Strategy. Standard ICP is sensitive to outlier correspondences, which are particularly problematic here because dynamic objects (persons, manipulated items) are present in the source but absent from the static target. These points will inevitably match to incorrect static surface regions. We employ **trimmed ICP** with fraction $\rho = 0.8$: at each iteration, only the closest 80% of correspondence pairs (ranked by Euclidean distance) are retained for the Kabsch fit. This discards the most distant 20% of correspondences at each iteration; it does not guarantee recovery when outliers or poor initialization dominate the retained matches.

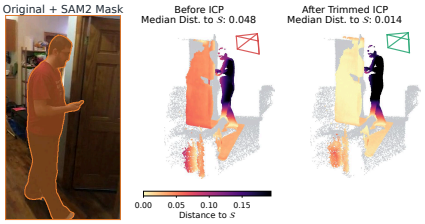


Figure 14: View 22 of 0DJ6R: the RGB frame with its SAM2 mask (left) and its dynamic points, colored by distance to $\mathcal{S}$, before and after trimmed ICP.

Alignment on 0DJ6R. Figure 14 and Stage (iv) of Figure 9 report trimmed ICP on all 40 dynamic frames of 0DJ6R, run with the parameters of Algorithm 2. Although the static and dynamic passes observe the same room, the two independent reconstructions disagree by a mean rigid offset of 0.058 scene units (≈13 cm at the metric scale recovered in Section D.1.3) and 2.3° (range 0.9°–5.0°). For the frame shown (view 22, whose RGB frame appears on the left of Figure 14), registration reduces the median distance from the dynamic points to $\mathcal{S}$ from 0.048 to 0.014 units. Across frames, the trimmed MSE drops by 59% on average (37–70%), and ICP converges after a median of 7 iterations (at most 18), well below $I_{\max}$. The actor's points have no counterpart in $\mathcal{S}$ and remain far from it after registration. In the frame shown, 98% of the correspondences discarded by the $\rho = 0.8$ trim belong to the actor (the masked region of the RGB frame), so trimming removes precisely the dynamic content that would otherwise bias the rigid fit.

Camera Pose Refinement. The ICP-derived rigid transform $\mathbf{T}_{\text{icp}} \in \mathrm{SE}(3)$ is composed with the π^3-predicted camera pose to produce a refined pose aligned to the static reference frame. For camera-to-world (c2w) and world-to-camera (w2c) poses, the updates are

$$\mathbf{T}_t^{\text{refined}} = \mathbf{T}_{\text{icp}} \cdot \mathbf{T}_t^{\text{dyn}} \quad \text{(c2w)}, \qquad \mathbf{E}_t^{\text{refined}} = \mathbf{E}_t^{\text{dyn}} \cdot \mathbf{T}_{\text{icp}}^{-1} \quad \text{(w2c)}. \tag{13}$$

This follows from the fact that points in the dynamic world frame are transformed as $\mathbf{p}' = \mathbf{T}_{\text{icp}} \cdot \mathbf{p}$, so the camera-to-world mapping must be left-multiplied by the same transform.

C.5 Mask-Aware Scene Merging

To construct the final 4D scene, we use precomputed interaction segmentation masks to selectively merge dynamic content with the static background. Per-frame binary interaction masks $\mathbf{M}_t \in \{0,1\}^{H\times W}$ are loaded from pre-computed segmentation outputs (from the SAM2 segmentation pipeline). For 0DJ6R, these masks cover the actor and occupy 18–25% of the frame (Figure 9, Final Output).

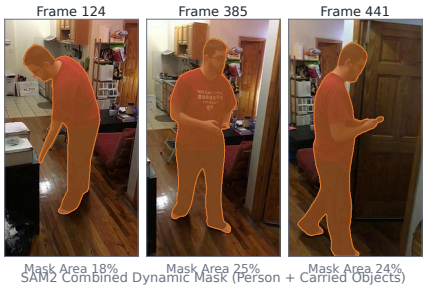


Figure 15: Pre-computed SAM2 masks that select the foreground merged into $\mathcal{S}$.

Table 7: **Hyperparameters for the 3D scene construction pipeline.**

Configuration Key	Value	Description & Role
Model Architecture		
Backbone	DINOv2-L (ViT-L/14)	Frozen visual encoder with register tokens
Decoder depth	36 layers	Alternating intra- and cross-view attention
Positional encoding	RoPE2D (f=100)	2D rotary position embeddings
Precision	`bfloat16` / `fp16`	Native mixed-precision forward pass
Filtering & Alignment		
Frame sampling	$\tau = 0.95$	Adaptive SIFT/homography overlap; annotated frames retained
τ_{static}	0.1	Confidence threshold for static scene
τ_{frame}	0.01	Confidence threshold for dynamic objects
Depth-edge rtol	0.03	Relative tolerance for edge discontinuity suppression
ICP iterations	100 (max)	Iterative closest point with early stop ϵ=10^{-5}
Trim fraction ρ	0.8	Correspondence trimming (closest 80%)
Voxelization & Intrinsic Geometry		
Dynamic voxel	0.01	Per-frame voxel size in reconstruction units
Merge voxel	0.02	Static–dynamic overlap voxel size in reconstruction units
Vertical FOV	0.96 rad ($\approx 55^\circ$)	Pinhole camera intrinsic vertical field of view

Point Partitioning. For each frame t, the dynamic frame's valid points are split into two sets: (1) $\mathcal{D}_t^{\text{all}}$: All valid points (used for ICP alignment). (2) $\mathcal{D}_t^{\text{fg}} = \{k \in \mathcal{D}_t^{\text{all}} : \mathbf{M}_t(\text{pixel}(k)) > 0\}$: Foreground-only points (used for merging).

Selective Merging Procedure. The merging follows a two-stage approach:

1. **ICP alignment (full frame)**: The ICP registration (Section C.4) uses *all* valid dynamic points $\mathcal{D}_t^{\text{all}}$ in frame t, including both foreground and background, to estimate the rigid

transform $\mathbf{T}_{\text{icp}}^{(t)}$. Using the full frame makes background correspondences available to the fit; trimming reduces the influence of large foreground residuals.

2. **Foreground-only merging**: Only the masked foreground points $\mathcal{D}_t^{\text{fg}}$ are transformed by $\mathbf{T}_{\text{icp}}^{(t)}$ and concatenated with the static background. This prevents duplication of background geometry (which already exists in the static cloud) while placing detected objects into the static reference frame.

We then obtain the final merged point cloud for frame t (Eq. (14)) and apply voxel deduplication.

$$\mathcal{P}_t = \mathcal{S} \cup \left\{ \mathbf{T}_{\text{icp}}^{(t)} \cdot \mathbf{d}_k^t \;\middle|\; k \in \mathcal{D}_t^{\text{fg}} \right\} \tag{14}$$

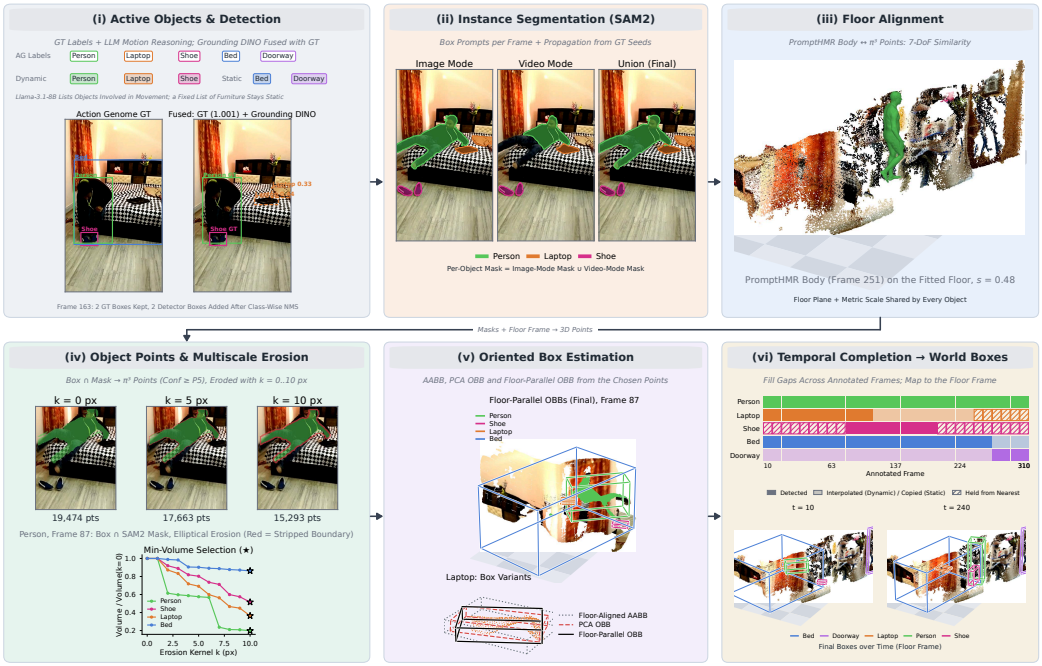


Figure 16: **3D bounding box pipeline on the ActionGenome video `00T1E`, shown with the actual intermediate outputs of each stage.** *(i)* The GT labels and the LLM's motion reasoning split the active objects into a dynamic set (person, laptop, shoe) and a static set (bed, doorway). On frame 163, GT boxes (pseudo-score 1.001) are fused with Grounding DINO proposals, which add two laptop candidates. *(ii)* SAM2 image-mode and video-mode masks on frame 87, and their per-object union. *(iii)* The PromptHMR body on the fitted floor, which fixes the floor plane and the metric scale (the body is displayed with a floor-preserving re-fit). *(iv)* Box-and-mask points of the person, eroded with elliptical kernels of size k (red marks the stripped boundary). The candidate volume decreases with k for every object, so minimum-volume selection keeps the most eroded candidate ($\star$). *(v)* The final floor-parallel OBBs on frame 87 and, for the laptop, the floor-aligned AABB, the PCA OBB, and the floor-parallel OBB fitted to the same points; only the floor-parallel OBB is stored. *(vi)* Per-label completion over the 29 annotated frames (detected, interpolated or copied, and held), followed by the final boxes in the floor frame at two timestamps.

D Geometric Annotation

We first present the automated pipeline that estimates world-frame 3D bounding boxes from the reconstructed point clouds of Section C (Section D.1). We then present the manual correction pipeline, which provides interactive tools to correct the floor transform and the estimated boxes (Section D.2).

D.1 Automated 3D Bounding Box Pipeline

This stage converts the reconstructed point clouds into world-frame **oriented 3D bounding boxes (OBBs)** for all interacting objects in ActionGenome Ji et al. (2019) video clips. RGB frames and existing 2D annotations identify the objects; segmentation isolates their pixels; the reconstruction supplies the corresponding 3D points. The pipeline has five components: (i)**Detection:** object detection via Grounding DINO Liu et al. (2024) with fusion with ground-truth annotations, (ii) **Segmentation:** instance segmentation via SAM2 Ravi et al. (2024) in dual image-and-video mode, (iii) **Object Classification:** static/dynamic object classification via LLM-based reasoning, (iv) **Object Point Cloud Determination:** floor-aligned 3D point extraction with multiscale mask erosion, and (v) **OBB Estimation:** floor-parallel oriented bounding box fitting with a minimum-area rectangle on the floor plane, followed by temporal completion across frames. Figure 16 illustrates every stage on the ActionGenome video `00T1E` with the actual intermediate outputs of the pipeline, and we refer to its panels throughout this section.

D.1.1 Object Detection

The detection stage produces per-frame, per-object 2D bounding boxes by fusing model-based detections with ground-truth annotations.

Grounding DINO Detection. We employ Grounding DINO Liu et al. (2024) as a zero-shot, open-vocabulary detector. Each frame is prompted with the video's active object labels (e.g., "`person. chair. cup.`") using a box threshold of 0.25 and text threshold of 0.3. Output labels are normalized to a canonical vocabulary.



Figure 17: AG ground truth (left) and the fused GT + Grounding DINO boxes (right).

Active Object Labels & Static/Dynamic Classification. Active object categories are determined from two sources: (a) labels extracted directly from ActionGenome ground-truth annotations, and (b) labels produced by an LLM-based reasoning pipeline using LLaMA Touvron et al. (2023). The LLM (Llama-3.1-8B-Instruct) identifies the objects involved in movement by extracting interactions from video captions and mapping them to the candidate label set via synonym-aware matching. The remaining GT active objects form the *static* set. This split drives two separate detection-segmentation pipelines: a static pipeline for background elements and a dynamic pipeline for actively used objects. For `00T1E`, it yields the dynamic objects person, laptop, and shoe and the static objects bed and doorway (Figure 16, panel (i)).

Ground Truth Annotation Fusion. To ensure annotated objects are not missed (especially small or occluded items), GT bounding boxes are assigned a pseudo-score of 1.001 and concatenated with Grounding DINO predictions. On frame 163 of `00T1E` (Figure 17), the fused set keeps the GT person and shoe boxes and adds two laptop proposals that the GT does not contain for this frame.

D.1.2 Instance Segmentation via SAM2

Given per-frame detected bounding boxes, we produce per-object binary segmentation masks using SAM2 Ravi et al. (2024) (hiera-base-plus) in two complementary modes. (a) **Image Mode.** SAM2 processes each frame independently: per-label bounding boxes are passed as box prompts in single-mask mode, and the resulting per-label masks are combined via boolean OR to produce one union mask per frame. (b) **Video Mode.** SAM2's video predictor propagates masks temporally from seed frames. Seeds are selected from annotated (GT) frames when available (up to 10 per label, spaced $\geq$10 frames apart, prioritized by detection score); otherwise, the first detection of each label is used. A single propagation pass produces masks for all seeded objects across all frames, binarized at threshold 0.5.

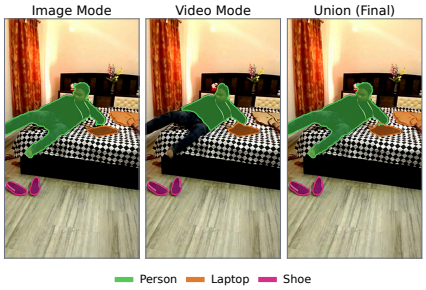


Figure 18: SAM2 image-mode, video-mode, and union masks on frame 87 of `00T1E`.

Mask Union. The final per-object mask is the pixel-wise union of image-mode and video-mode masks, $\mathbf{M}_{\text{final}}^{(l,t)} = \mathbf{M}_{\text{image}}^{(l,t)} \vee \mathbf{M}_{\text{video}}^{(l,t)}$, where l denotes the object label and t the frame index. A pixel is retained if either mode assigns it to the object. The union guards against failures of either mode. On frame 87 of `00T1E` (Figure 18), the image-mode mask covers the full body of the reclining person, whereas the propagated video-mode mask covers only the upper body. The union therefore retains the full silhouette, and an object is lost only when both modes miss it.

D.1.3 Floor Alignment via SMPL-Based Similarity Estimation

Before computing 3D bounding boxes, we establish a floor-aligned coordinate system by registering the π^3-reconstructed point cloud to a canonical frame where the floor plane is horizontal.

SMPL Mesh Correspondence. We use estimated SMPL Loper et al. (2015) body meshes as anchors for scale-and-pose recovery. For each frame, 2D body keypoints (from OpenPose Cao et al. (2017)) are paired with fitted SMPL 3D joint positions (in metric space), yielding K correspondence pairs $\{(\mathbf{s}_k, \mathbf{d}_k)\}$ between SMPL points and π^3 scene points. For each frame with $K \geq 3$ correspondences, a 7-DoF similarity transformation (scale s, rotation $\mathbf{R}$, translation $\boldsymbol{\tau}$) mapping the metric SMPL space onto the scene is estimated via RANSAC:

$$\mathbf{d}_k \approx s \cdot \mathbf{R} \cdot \mathbf{s}_k + \boldsymbol{\tau} \tag{15}$$

Global Floor Transform. Per-frame similarity estimates are robustly averaged to produce a single global similarity $(s_{\text{avg}}, \mathbf{R}_{\text{avg}}, \boldsymbol{\tau}_{\text{avg}})$, requiring at least $\max(3, 20\%)$ valid frames. Frames whose scale or translation deviates from the median by a median-absolute-deviation test, or whose rotation angle exceeds $15°$, are discarded first. The remaining estimates are averaged with weights given by their number of correspondences, using weighted quaternion averaging for the rotations. In the floor-aligned frame, the Y-axis is the floor normal (up) and the XZ plane is the floor surface. The floor-alignment transform maps world points to floor-aligned (metric) coordinates:

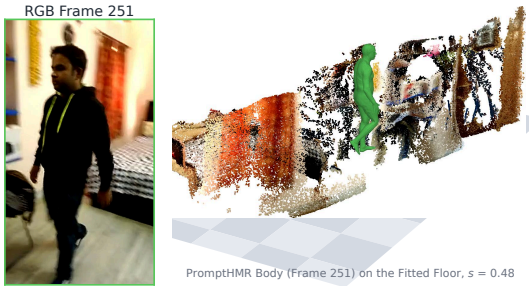


Figure 19: RGB frame 251 of `00T1E` (left) and the PromptHMR body of that frame on the fitted floor (right; displayed with a floor-preserving re-fit).

$$\mathbf{p}_{\text{floor}} = \frac{1}{s_{\text{avg}}} \cdot \mathbf{R}_{\text{avg}}^{\top} \cdot (\mathbf{p}_{\text{world}} - \boldsymbol{\tau}_{\text{avg}}) \tag{16}$$

and its inverse recovers world coordinates:

$$\mathbf{p}_{\text{world}} = s_{\text{avg}} \cdot \mathbf{R}_{\text{avg}} \cdot \mathbf{p}_{\text{floor}} + \boldsymbol{\tau}_{\text{avg}} \tag{17}$$

Figure 19 shows the resulting floor and the PromptHMR body for `00T1E`, for which $s_{\text{avg}} = 0.48$.

D.1.4 Oriented 3D Bounding Box Computation

For each interacting object in each frame, we extract masked 3D points from the π^3 reconstruction and enclose them with a floor-parallel OBB, whose vertical axis coincides with the floor normal and whose only rotation is the yaw about that normal. For reference, we also fit an unconstrained PCA-based OBB to the same points; this box is not used in the annotations.

Masked 3D Point Extraction. The 2D bounding box (GT or Grounding DINO) is resized to the π^3 prediction resolution and intersected with the corresponding SAM2 mask (Section D.1.2) to produce a tight object mask (falling back to the rectangular bbox if no mask exists). For objects discovered only by Grounding DINO and absent from the GT, the resized rectangle is used without a SAM2 mask. Points are then filtered by the per-pixel confidence map (thresholded at the frame's 5th-percentile, clamped $\geq 10^{-3}$) and by finite-coordinate validity.



Figure 20: Box ∩ mask points of the person (frame 87) under elliptical erosion; red marks the stripped boundary.

Multiscale Erosion. To strip boundary artifacts from imprecise masks, we erode the object mask with elliptical kernels of size $k \in \{0, 1, \ldots, 10\}$ px, extract the surviving 3D points at each level (requiring $\geq$50 points), and select the erosion yielding the **minimum-volume** bounding box. Because erosion only removes pixels, the surviving point set shrinks as k grows, so the floor-aligned volume below is non-increasing in k. The selection therefore returns the most eroded candidate that retains at least 50 points. On `00T1E`, all 80 object instances select $k = 10$. For the person in frame 87 (Figure 20), erosion keeps 15,293 of 19,474 points (79%) yet shrinks the box volume by 80%. This

indicates that the stripped boundary pixels carry the outlying points that inflate the box. Candidate volumes are computed from coordinate-wise extents in floor-aligned coordinates (Eq. (16)):

$$V = \prod_{d \in \{x,y,z\}} (\max_k p^{\text{floor}}_{k,d} - \min_k p^{\text{floor}}_{k,d}) \tag{18}$$

From the filtered 3D points of the selected erosion scale, we compute two boxes: a PCA OBB, which serves only as a reference, and the floor-parallel OBB, which is stored in the annotations.

PCA OBB (Reference). The reference box has a full 3-DoF rotation obtained via PCA on the object point cloud: the covariance matrix $\mathbf{C} = \frac{1}{N-1}\sum_k (\mathbf{p}_k - \boldsymbol{\mu})(\mathbf{p}_k - \boldsymbol{\mu})^\top$ is eigen-decomposed as $\mathbf{C} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^\top$, points are projected onto the principal axes $\mathbf{q}_k = \mathbf{V}^\top(\mathbf{p}_k - \boldsymbol{\mu})$, and extents are taken as $e_d = \max_k q_{k,d} - \min_k q_{k,d}$. The OBB center is adjusted for asymmetric distributions:

$$\mathbf{c}_{\text{obb}} = \boldsymbol{\mu} + \mathbf{V} \cdot \frac{\mathbf{q}_{\min} + \mathbf{q}_{\max}}{2} \tag{19}$$

Eight corners are generated from $\pm e_d/2$ along principal axes and rotated back to world coordinates via $\mathbf{V}$. Because its principal axes follow the directions of largest spread in the partial point cloud rather than gravity, the PCA OBB is rarely upright: over 150 randomly sampled training videos, its axis closest to the floor normal deviates from the normal by a median of 25°.

Floor-Parallel OBB. The floor-parallel OBB permits only yaw rotation around the floor normal: points are transformed to the floor-aligned frame (Eq. (16)), projected onto the XZ plane, and enclosed by the minimum-area 2D rectangle, which fixes the yaw. The rectangle is extruded vertically over $[y_{\min}, y_{\max}]$ to produce 8 corners, which are transformed back to world coordinates via Eq. (17). The floor-parallel OBB is the only box stored in the annotations. We use it for every object, static or dynamic, including the objects detected only by Grounding DINO, and no later stage reads the PCA OBB, which remains only in the intermediate per-frame outputs. Figure 21 shows the three boxes for the laptop, and panel (v) of Figure 16 shows the final boxes of frame 87. The PCA OBB tilts with the laptop's surface, whereas the floor-parallel OBB keeps a vertical axis that is consistent across objects and frames. These automatically fitted boxes initialize the training geometry in the final frame of Section D.1.6; available manual floor corrections are incorporated by the corrected-geometry pipeline, without implying exhaustive manual review of training boxes. For the test split, the boxes are refit after manual floor correction (Section D.2.1); the refit estimates the yaw with a 2D PCA instead of the minimum-area rectangle but keeps every box floor-parallel.

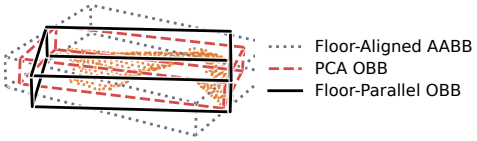


Figure 21: Laptop points (frame 87) with the floor-aligned AABB, PCA OBB, and floor-parallel OBB.

D.1.5 Temporal Completion

Boxes are estimated only on the frames in which an object is detected, so the final 4D annotation completes each object's track over all annotated frames. For dynamic objects, the box center and extents are linearly interpolated, and the rotation is spherically interpolated, between neighboring detections; before the first and after the last detection, the nearest box is held. For static objects, a frame without a detection copies the box of the nearest earlier detection (or of the first detection). Per-frame minimum-volume erosion selection (Section D.1.4) keeps the box scale consistent across frames.

Figure 22 shows the completion for `00T1E` together with the resulting world boxes. The person is detected in all 29 annotated frames. The laptop is detected in 12

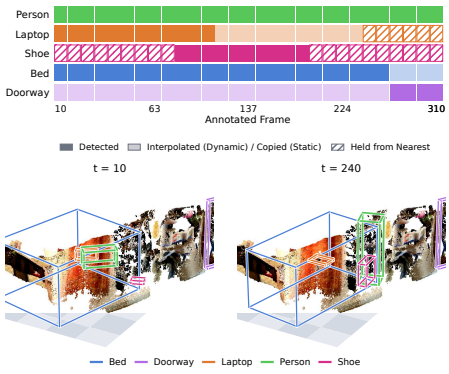


Figure 22: Per-label completion over the annotated frames (top) and the boxes in the floor frame (bottom).

frames, 11 frames are interpolated, and its box is held over the final 6. The shoe is detected in 10 consecutive frames and held in the remaining 19. The doorway is visible only at the end of the clip; it is detected in 4 frames and copied to the other 25.

D.1.6 WORLD-TO-FINAL COORDINATE TRANSFORM

For downstream processing, all annotations are converted from the π^3 world frame to a **final coordinate system** that is floor-aligned (XY = floor, Z = up) and centered at the floor origin. The world-to-final transform is the rigid map:

$$\mathbf{p}_{\text{final}} = \mathbf{A} \cdot (\mathbf{p}_{\text{world}} - \mathbf{o}_{\text{world}}) \tag{20}$$

where $\mathbf{A} = \mathbf{M}\,[\mathbf{r}_1 \;\; \mathbf{r}_3 \;\; \mathbf{r}_2]^\top$ is built from the columns $\mathbf{r}_i$ of $\mathbf{R}_{\text{avg}}$, $\mathbf{M} = \text{diag}(-1, 1, 1)$ keeps the frame right-handed, and $\mathbf{o}_{\text{world}} = \boldsymbol{\tau}_{\text{avg}}$. Since $\mathbf{A}$ is a rotation, final-frame coordinates retain the scale of the π^3 reconstruction. The similarity scale s_{avg} maps meters to reconstruction units and is stored separately; in this automatic final frame, meter-valued coordinates are $\mathbf{p}_{\text{meters}} = \mathbf{p}_{\text{final}}/s_{\text{avg}}$. The stored corners and camera translations, and the geometry consumed by the supervised models and evaluator, remain in reconstruction units. Manual floor alignment applies a further rigid transform to scene geometry, as detailed in Section D.2.1; it does not convert the stored scene coordinates to meters. Camera poses are similarly transformed:

$$\mathbf{R}_{\text{final}} = \mathbf{A} \cdot \mathbf{R}_{\text{world}}, \quad \boldsymbol{\tau}_{\text{final}} = \mathbf{A} \cdot (\boldsymbol{\tau}_{\text{world}} - \mathbf{o}_{\text{world}}) \tag{21}$$

D.2 MANUAL CORRECTION PIPELINE

The manual correction pipeline provides two interactive tools that follow the order of the automated pipeline: a floor-correction tool that fixes the gravity-aligned frame (Section D.2.1), and the WorldBBox Viewer for correcting the world-frame boxes in that frame (Section D.2.2). Section D.2.3 summarizes the reported agreement measures.

D.2.1 CORRECTING THE FLOOR TRANSFORM

Feed-forward monocular 3D reconstruction methods (e.g., DUSt3R Wang et al. (2024b) and the π^3 Wang et al. (2025c) model used in our pipeline) produce a world frame whose vertical axis need not coincide with gravity. We transform the reconstructed scene into a *gravity-aligned reference frame* (XY-plane = floor, $+Z$ = up) via three stages: (i) *automatic floor alignment* from the reconstruction's floor mesh and global similarity transform (Sections D.1.3 and D.1.6), (ii) *manual interactive correction* with 6-DoF controls, and (iii) *automated XY-plane alignment* that analytically maps the corrected floor to $Z{=}0$.

Automatic Floor Alignment via Global Floor Similarity. The correction starts from the automatic alignment. The reconstruction pipeline extracts a floor mesh $\mathcal{F} = (\mathbf{V}_0, \mathbf{F}_0, \mathbf{C}_0)$ with vertex positions $\mathbf{V}_0 \in \mathbb{R}^{N_v \times 3}$, faces $\mathbf{F}_0$, and optional colors $\mathbf{C}_0$, together with the global floor similarity $(s_g, \mathbf{R}_g, \boldsymbol{\tau}_g)$ of Section D.1.3. Floor vertices are transformed into the world frame as:

$$\mathbf{v}_{\text{world}} = s_g \cdot (\mathbf{v}_0 \cdot \mathbf{R}_g^\top) + \boldsymbol{\tau}_g, \quad \forall\, \mathbf{v}_0 \in \mathbf{V}_0. \tag{22}$$

From the columns of $\mathbf{R}_g$ we extract in-plane tangent directions $\boldsymbol{\ell}_1 = \mathbf{R}_g[:, 0]$, $\boldsymbol{\ell}_2 = \mathbf{R}_g[:, 2]$ and floor normal $\mathbf{n} = \mathbf{R}_g[:, 1]$, forming the basis $\mathbf{F} = [\boldsymbol{\ell}_1 \mid \boldsymbol{\ell}_2 \mid \mathbf{n}]$. The intermediate world-to-floor axis map is $\mathbf{R}_{\text{align}} = \mathbf{F}^\top$ with translation $\boldsymbol{\tau}_{\text{align}} = -\mathbf{R}_{\text{align}} \cdot \boldsymbol{\tau}_g$:

$$\mathbf{x}_{\text{floor}} = \mathbf{R}_{\text{align}} \cdot \mathbf{x}_{\text{world}} + \boldsymbol{\tau}_{\text{align}}. \tag{23}$$

Swapping the second and third axes gives $\det(\mathbf{R}_{\text{align}}) = -1$. The required axis-sign change $\mathbf{M} = \text{diag}(-1, 1, 1)$ restores a right-handed frame and yields the final rotation:

$$\mathbf{R}_{\text{final}} = \mathbf{M} \cdot \mathbf{R}_{\text{align}}, \quad \boldsymbol{\tau}_{\text{final}} = \mathbf{M} \cdot \boldsymbol{\tau}_{\text{align}}, \tag{24}$$

so that $\mathbf{x}_{\text{final}} = \mathbf{R}_{\text{final}} \cdot \mathbf{x}_{\text{world}} + \boldsymbol{\tau}_{\text{final}}$.

Manual Interactive Floor Correction. The automatic alignment is often imperfect. An interactive 3D tool renders the floor mesh, point cloud, and bounding boxes, allowing the annotator to adjust a correction transform $\mathbf{T}_\delta = (\mathbf{R}_\delta, \boldsymbol{\tau}_\delta, \mathbf{s}_\delta)$ using translation, rotation, and scale controls. This adjustment acts on the floor mesh; the point cloud and boxes remain fixed while the floor is placed against them. Both original (wireframe) and corrected (solid) floor meshes are displayed simultaneously for visual feedback. The tool saves the correction in Firebase for each video and reapplies it when the scene is reopened.

Automated XY-Plane Alignment. After manual correction the floor may still not lie in the canonical XY-plane. We compute $\mathbf{T}_{\text{XY}} = (\mathbf{R}_{\text{XY}}, \boldsymbol{\tau}_{\text{XY}})$ to map the corrected floor to $Z{=}0$ in six steps:

1. Apply the floor correction. Let $\mathbf{V}_{\text{auto}}$ be the floor vertices after the world and automatic final-frame transforms above. These vertices are transformed by the correction $\mathbf{T}_\delta$ (Section D.2.1):

$$\mathbf{v}_{\text{corr}} = \mathbf{R}_\delta \cdot (\mathbf{s}_\delta \odot \mathbf{v}_{\text{auto}}) + \boldsymbol{\tau}_\delta, \quad \forall\, \mathbf{v}_{\text{auto}} \in \mathbf{V}_{\text{auto}}. \tag{25}$$

2. Estimate the floor normal. The floor centroid $\bar{\mathbf{v}}$ and robust normal $\hat{\mathbf{n}}$ are estimated from triangle cross products (selecting the largest-magnitude result); the normal is flipped if needed so that $\hat{n}_z > 0$.

3. Align the normal with the vertical axis. The rotation $\mathbf{R}_{\text{norm}}$ aligning $\hat{\mathbf{n}}$ with $+Z$ is obtained via Rodrigues' formula with axis $\hat{\mathbf{k}} = \frac{\hat{\mathbf{n}} \times \hat{\mathbf{z}}}{\|\hat{\mathbf{n}} \times \hat{\mathbf{z}}\|}$ and angle $\phi = \arccos(\hat{\mathbf{n}} \cdot \hat{\mathbf{z}})$:

$$\mathbf{R}_{\text{norm}} = \cos\phi \cdot \mathbf{I} + \sin\phi \cdot [\hat{\mathbf{k}}]_\times + (1 - \cos\phi) \cdot \hat{\mathbf{k}}\hat{\mathbf{k}}^\top. \tag{26}$$

4. Correct the in-plane rotation. The residual in-plane misalignment is corrected by a Z-axis rotation $\mathbf{R}_z(\alpha)$ where $\alpha = -\text{atan2}(\tilde{x}_y, \tilde{x}_x)$ and $\tilde{\mathbf{x}} = \mathbf{R}_{\text{norm}} \cdot \mathbf{R}_\delta \cdot [1, 0, 0]^\top$. The combined rotation is $\mathbf{R}_{\text{XY}} = \mathbf{R}_z(\alpha) \cdot \mathbf{R}_{\text{norm}}$.

5. Translate the floor to the origin. The orthogonal projection of the origin onto the floor plane $\mathbf{p}_\cap = (\hat{\mathbf{n}} \cdot \bar{\mathbf{v}}) \cdot \hat{\mathbf{n}}$ is mapped to the origin: $\boldsymbol{\tau}_{\text{XY}} = -\mathbf{R}_{\text{XY}} \cdot \mathbf{p}_\cap$.

6. Store the rotation. $\mathbf{R}_{\text{XY}}$ is decomposed into Euler angles $(\theta_x, \theta_y, \theta_z)$ (ZYX convention) for storage.

Bounding Box Refitting. After applying $\mathbf{T}_{\text{XY}}$, we refit the boxes of the test split in the gravity-aligned frame from the points of each object. The point selection follows the automated pipeline (Section D.1.4): the object's SAM2 mask, intersected with its 2D bounding box, is eroded with elliptical kernels $k \in \{0, 1, \ldots, 10\}$ px, and the 2D box itself takes the place of the mask when no mask is available, as for static objects. Every scale that retains at least 50 confident points yields a candidate box with vertical edges along Z. The heading of the candidate is the principal axis of the 2×2 covariance $\boldsymbol{\Sigma}$ of the XY coordinates of its points,

$$\theta = \tfrac{1}{2}\,\text{atan2}(2\,\Sigma_{xy},\ \Sigma_{xx} - \Sigma_{yy})\,, \tag{27}$$

its XY extents are the ranges of the points projected onto $(\cos\theta, \sin\theta)$ and $(-\sin\theta, \cos\theta)$, its vertical extent is the range of their Z coordinates, and its center lies at the midpoints of these three ranges. The candidate with the smallest volume is kept. The refit boxes are therefore floor-parallel by construction; only their yaw comes from the 2D PCA, in place of the minimum-area rectangle used by the automated pipeline. When no scale retains enough points, we instead refit the points that fall inside the object's automatic box, identified with the separating axis test; this fallback affects 13 boxes of the test split.

Final Alignment Composition. The scene geometry and the floor mesh use different compositions. Scene points, boxes, and camera poses receive the rigid map

$$\mathbf{x}_{\text{canonical}} = \mathbf{T}_{\text{XY}} \circ \mathbf{T}_{\text{auto}}(\mathbf{x}_{\text{world}}), \tag{28}$$

where $\mathbf{T}_{\text{auto}}$ is from Eq. (24) and $\mathbf{T}_{\text{XY}}$ is derived from the corrected floor. Only floor vertices additionally receive the manual adjustment:

$$\mathbf{v}_{\text{canonical}} = \mathbf{T}_{\text{XY}} \circ \mathbf{T}_\delta \circ \mathbf{T}_{\text{auto}}(\mathbf{v}_{\text{world}}). \tag{29}$$

The manual floor scale therefore changes the reference mesh, not the physical scale of the scene points or boxes. The two composed transforms are cached per video; scene coordinates remain in reconstruction units.

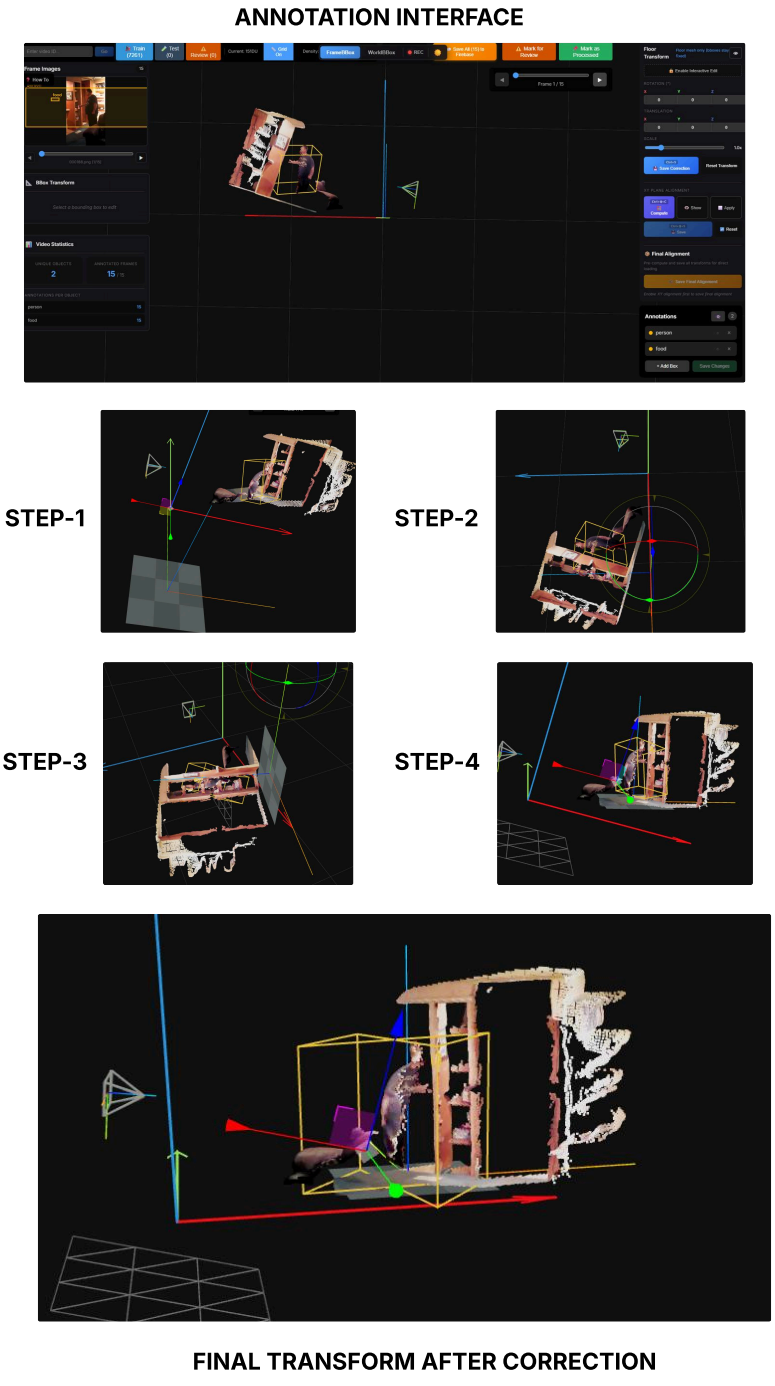


Figure 23: Overview of the annotation interface used for manual floor correction. The figure illustrates the sequential adjustment process across four steps, culminating in the final corrected transform to align the scene with the corrected floor.

D.2.2 CORRECTING WORLD-FRAME 3D BOUNDING BOXES

With the gravity-aligned frame established (Section D.2.1), the boxes refit in that frame (Section D.2.1) initialize the world-frame annotations. These automatic annotations often exhibit errors (incorrect dimensions, missing objects, misaligned orientations, wrong labels), so we provide the *WorldBBox Viewer* for interactive 3D correction.

Initial World-Frame Boxes. The viewer starts from world-level 4D annotations assembled from the refit boxes $\mathbf{C}_{\text{final}} \in \mathbb{R}^{8\times 3}$ (Section D.2.1), in which each object keeps its static or dynamic class (Section D.1.1). Each static label ℓ is represented in every annotated frame by a single box that encloses its refit boxes over all frames and has edges parallel to the axes of the gravity-aligned frame:

$$\mathbf{C}^{(\ell)}_{\text{union}} = \text{AABB}\left(\bigcup_{f \in \text{frames}} \mathbf{C}^{(\ell,f)}_{\text{final}} \right), \tag{30}$$

where $\text{AABB}(\cdot)$ spans the coordinate-wise minimum and maximum of the pooled corners. A static object therefore stays fixed across the clip, although its box does not follow the heading of the object. Dynamic objects keep their per-frame refit boxes and are not filled at this stage.

Interactive Correction Interface. The WorldBBox Viewer provides a 3D canvas (via Three.js/ React Three Fiber) rendering the floor mesh, reconstructed point cloud, bounding box wireframes with floating text labels, and relationship arcs connecting subject-predicate-object triples. The viewer supports orbital camera controls and frame-by-frame navigation. For each selected bounding box, the annotator can apply translation, rotation, and scaling using sliders with a live preview. Transforms are applied sequentially (scale $\rightarrow$ rotate $\rightarrow$ translate about box center) and committed on confirmation, with an undo stack for reverting. The annotations panel additionally supports inline label editing, per-box visibility toggles, and adding/deleting boxes.

Temporal Propagation of Annotations. A key challenge in world-level annotation is ensuring that object bounding boxes are consistent across all frames. The Propagation Panel provides tools to copy a bounding box annotation from the current frame to selected target frames. The annotator specifies (1) **Direction**: Forward (to subsequent frames), backward (to preceding frames), or both. (2) **Frame Count**: A specific number of frames to propagate to, or "All" to propagate to the video's start/end. Upon clicking **Propagate**, the selected box's label, corners, and center are copied to each target frame.

Export. When the world-frame boxes are exported and merged with the relationship annotations, an object that has no box in a frame takes its box from the nearest annotated frame in which it has one. A plane is then fitted to the corrected floor mesh, and the rigid transform that maps it onto $Z{=}0$ is applied to all boxes and camera poses; the rotation is applied only when the fitted floor deviates from the XY-plane by less than 10°. Consequently, the vertical axes of all test boxes lie within 0.5° of the floor normal.

D.2.3 GEOMETRIC AGREEMENT

The reported automatic-proposal mean 3D IoU is 0.385, while corrected boxes have reported inter-annotator 3D IoU above 0.9. These compare different stages and references: inter-annotator agreement measures consistency, not independently calibrated metric accuracy. Shared reconstruction or scale errors may remain even when annotators agree.

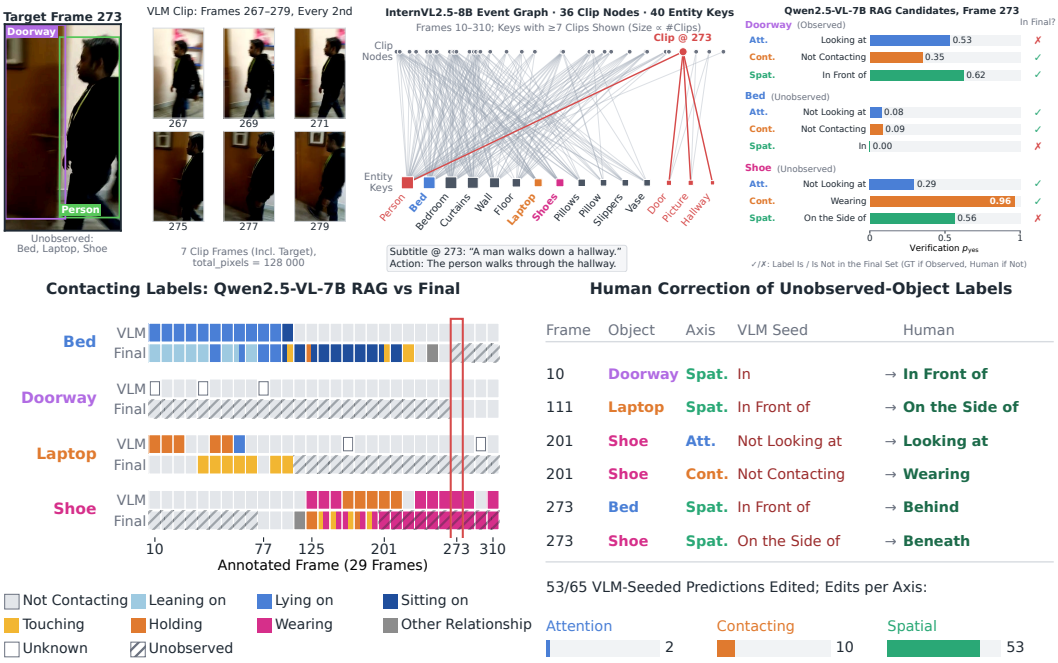


Figure 24: **Semantic annotation on the ActionGenome video `00T1E`, shown with the actual intermediate outputs.** *Top, left to right:* target frame 273 and the seven-frame clip that the VLM receives, in which the bed, laptop, and shoe are unobserved; the coarse event graph built by InternVL 2.5 (36 clip nodes, 40 entity keys), with the clip containing frame 273 highlighted; and the Qwen2.5-VL RAG candidates for frame 273 with their verification scores p_{yes}, marked by whether each label is in the final annotation. *Bottom:* contacting labels over the 29 annotated frames before and after correction, with unobserved frames hatched; and examples of human corrections of the VLM-seeded labels (53 of 65 seeded predictions were edited, mostly on the spatial axis).

E Semantic Annotation

The semantic annotation stage assigns person–object relationship labels at every annotated timestamp, including timestamps at which the object is unobserved. We first present the automated pipeline (Section E.1), which uses vision-language models (VLMs) to generate candidate labels and then verifies those labels automatically. The candidates provide pseudo-labels for training. We then present the manual correction pipeline (Section E.2), in which annotators review and correct these candidates to produce the test-split reference labels. Figure 24 traces both pipelines on the ActionGenome video `00T1E` with their actual intermediate outputs, which we use as a running example.

We retain the ActionGenome annotation schema Ji et al. (2019): one attention label, one or more contacting labels, and one or more spatial labels per person–object pair. Section B.2 lists the complete vocabulary. The bottom-left panel of Figure 24 shows the contacting labels of `00T1E` over its 29 annotated frames, as generated by Qwen2.5-VL with RAG and after correction. Hatched cells mark frames in which the object is unobserved, for which the stage must still produce labels.

E.1 Automated Relationship Annotation Pipeline

Here we describe how the candidates and their confidence scores are produced: the VLMs (Section E.1.1), the generation and verification setup (Section E.1.2), and the retrieval-augmented annotation method (Section E.1.3).

E.1.1 Models

We employ three open-source VLMs from different model families and with different parameter counts, all deployed behind the vLLM inference engine for efficient batched generation with tensor parallelism: (1) **Kimi-VL** (Kimi-VL-A3B-Instruct) - a lightweight mixture-of-experts VLM; (2)

InternVL 2.5 (InternVL2_5-8B) - a mid-scale VLM; (3) **Qwen 2.5-VL** (Qwen2.5-VL-7B-Instruct) - an instruction-tuned VLM from the Qwen family. All models support variable-length multi-frame video input. A BGE embedding model (bge-large-en-v1.5) is additionally used for text-similarity computations in the RAG-based annotation method.

E.1.2 Generation Setup

Video Input. Video frames are loaded from disk and converted to a $T \times C \times H \times W$ float tensor. A pixel-budget constraint (`total_pixels`$= 128000$) is enforced to keep memory consumption manageable. For each annotated frame, we supply a clip from its temporal neighborhood so that the model can use nearby observations when predicting the target-frame relationships. For target frame 273 of `00T1E` (Figure 25), the clip covers frames 267–279 at every second frame. Only the person and the doorway are visible in the target frame, whereas the bed, laptop, and shoe are unobserved and must be reasoned about from context.

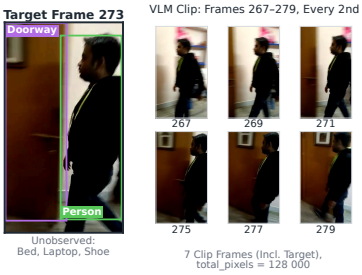


Figure 25: Target frame 273 of `00T1E` and its seven-frame clip.

Relationship Query Prompt. Let $\mathcal{O}_v$ be the video-level object inventory. For each $o \in \mathcal{O}_v$ and each annotated frame, a structured prompt elicits predictions for all three relationship axes:

> *You are analyzing a video clip from a scene. The object "{o}" is one of the objects present in this video. It may or may not be visible in the current frame. A person IS visible in this frame.*
> *Based on the full video context, predict the relationships between the person and the object "{o}".*
> 1. ATTENTION: Pick EXACTLY ONE from: {attention labels}
> 2. CONTACTING: Pick ONE OR MORE from: {contacting labels}
> 3. SPATIAL: Pick ONE OR MORE from: {spatial labels}
>
> *Respond ONLY in JSON format...*

Discriminative Verification. After the generative prediction step, each predicted relationship label is independently verified through a discriminative Yes/No classification using the same VLM. For a predicted relationship (o, r), the verification prompt varies by axis:

> (1) **Attention Relationship**: "*In this video, is the person {r} the {o}? Answer only Yes or No.*"; (2) **Contacting Relationship**: "*In this video, is the person {r} the {o}? Answer only Yes or No.*"; (3) **Spatial Relationship**: "*In this video, is the {o} {r} the person? Answer only Yes or No.*"

All verification prompts across all frames and objects are collected and processed in batches of up to 64 prompts, yielding a confidence score p_{yes} for every predicted label. The score is the probability of *Yes* normalized over *Yes* and *No*, computed from the first-token log-probabilities. Figure 26 shows the candidates for frame 273 of `00T1E`. Visibility does not determine confidence: the unobserved shoe receives $p_{\text{yes}} = 0.96$ for *wearing*, which the final annotation retains, whereas the visible doorway receives only 0.53 for *looking at*, which the final annotation rejects. The pipeline applies no threshold on p_{yes} and discards only unknown or out-of-vocabulary labels, so the scores act as confidence estimates rather than as a filter.

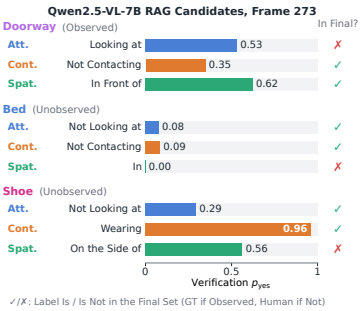


Figure 26: Qwen2.5-VL candidates for frame 273 of `00T1E` with their verification scores p_{yes}.

E.1.3 Annotation Methods

The primary annotation method retrieves video context before generating a frame-specific answer.

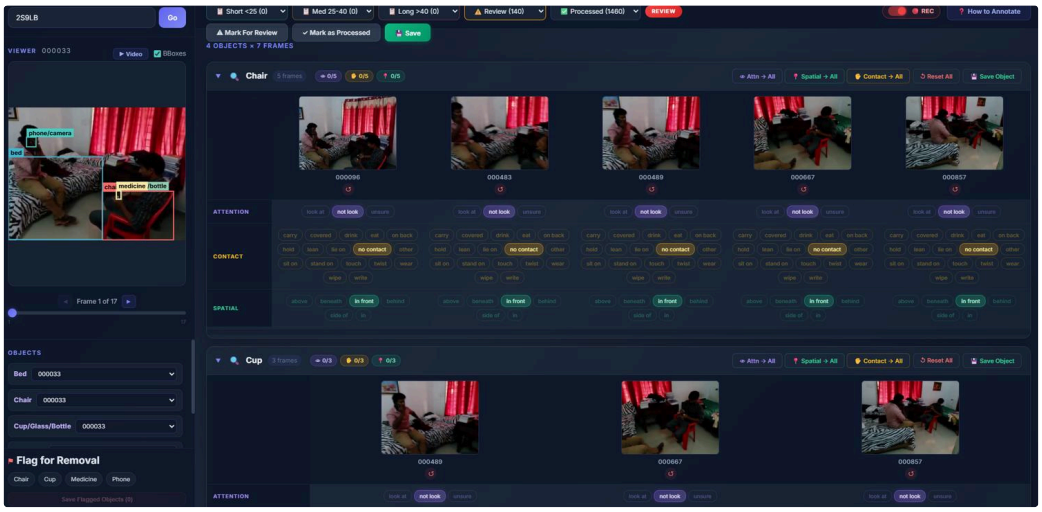


Figure 28: Interface for manual relationship correction in WSGG. The tool presents object-centric frame sequences and allows annotators to refine attention, contacting, and spatial predicates across time for accurate scene graph construction.

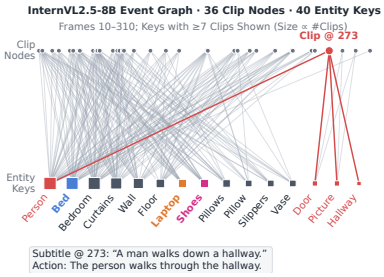


Figure 27: Coarse event graph of `00T1E` built by InternVL 2.5; the clip containing frame 273 is highlighted.

RAG Generation. We use retrieval-augmented generation (RAG) over the precomputed coarse event graph (Figure 4 in the main paper). Per-clip graphs are merged into a video-level directed graph, and a four-step pipeline produces relationship predictions: (1) keyword extraction from query prompts via a batched VLM call; (2) cached node retrieval using BGE embeddings (mean cosine similarity over the extracted keywords > 0.5); (3) template-based node refinement via batched Yes/No verification; (4) frame-specific answer generation using annotation-driven clips with graph-node context prepended to the prompt. Queries are processed and deduplicated across frames, and all batched calls are chunked into sub-batches of 64. For `00T1E`, the merged InternVL 2.5 graph links 36 clip nodes to 40 entity keys (Figure 27). The clip containing frame 273 is summarized as "A man walks down a hallway" and connects to the *person*, *door*, *picture*, and *hallway* keys.

E.2 Manual Correction Pipeline

The pseudo-annotations produced by the automated pipeline contain errors from hallucination, ambiguous visual context, or incorrect grounding. To create the test-set reference labels, trained annotators review these proposals in a purpose-built web annotation tool. They systematically review and correct every pseudo-annotated relationship across three semantic axes: **Attention**, **Contacting**, and **Spatial**. The workflow refines the proposed labels rather than starting from an empty annotation. Section E.2.2 distinguishes the measured correction activity from evidence about residual bias.

E.2.1 Annotation Interface

The tool follows a two-panel layout. The *left panel* provides a high-fidelity frame viewer with:

- Full-resolution frame display with a slider for scrubbing through all video frames.
- An HTML5 canvas overlay rendering person bounding boxes (green) and object bounding boxes (15 distinct colors) with class-name labels.
- A lightbox modal for pixel-level inspection with zoom (0.5–5×) and pan.

The *right panel* serves as the annotation workspace:

- **Video selection**: three dropdown selectors group videos by status (Pending / Review / Processed).

Table 8: **Label Flip Rate (LFR) in human correction logs.** Rates evaluate annotator corrections against automated VLM proposals over 61,042 reviewed instances per semantic axis. High editing rates, particularly on spatial relationships (65.4%), demonstrate rigorous manual intervention rather than passive acceptance of pseudo-labels.

Axis	Instances	Unchanged	LFR (%)	Selected Per-Class Flip Rates
Attention	61,042	51,480	15.7%	`looking_at`: 50.3%; `not_looking_at`: 15.3%
Contacting	61,042	49,903	18.2%	`sitting_on`: 75.2%; `holding`: 71.8%; `lying_on`: 70.8%; `carrying`: 64.5%; `not_contacting`: 16.9%
Spatial	61,042	21,115	65.4%	`behind`: 80.6%; `above`: 79.0%; `in`: 71.8%; `on_the_side_of`: 68.5%; `in_front_of`: 62.7%; `beneath`: 1.8%

- **Object accordions**: each unobserved object is presented as a collapsible section showing the object name, frame count, and per-category correction statistics.
- **Annotation grid**: a table of frames $\times$ relationship categories. Each cell contains toggle buttons for every possible label. Attention is single-select; Contacting and Spatial are multi-select. Active labels are color-coded with abbreviated display names.

E.2.2 Annotation Agreement and Correction Activity

The training and test semantic annotations have different provenance. Original visible relationships use AG labels. Unobserved training relationships have VLM pseudo-labels, used by the W-baselines but excluded by the reference WorldWise recipe. Unobserved test relationships undergo human correction. The reported agreement study covers 200 videos and gives 92%, 90%, and 89% agreement for attention, spatial, and contacting, respectively. These percentages characterize agreement under the annotation procedure; they are not independent estimates of residual anchoring bias.

Label Flip Rate. Table 8 reports changes between the original VLM proposals and final corrected labels in the available correction logs. Spatial labels were changed in 65.4% of reviewed instances, compared with 15.7% for attention and 18.2% for contacting. Selected positive predicates also have high reported flip rates: `behind` (80.6%), `sitting_on` (75.2%), `holding` (71.8%), and `looking_at` (50.3%). These rates show substantial annotation editing, especially for spatial relationships. They do not measure the error remaining after correction or demonstrate that unchanged proposals are correct.

F SUPERVISED WSGG METHODS

F.1 SHARED COMPONENTS

The eight supervised WSGG methods of this appendix, the five adapted video scene graph baselines (Appendix F.2) and the three **WORLDWISE** models (Appendix F.3), are assembled from one set of components and trained with one procedure. This subsection describes those components once, unit by unit, together with the shared losses and the shared training procedure. The method subsections that follow state which components a method instantiates, describe what it adds, and give its forward pass and training objective. Table 9 lists the modules of every method and Table 10 the configuration of every method.

F.1.1 TASK, INPUTS AND THE FOUR PROCESSING UNITS

Task and I/O contract. Every supervised WSGG method solves the same problem on the same tensors. A video is represented by T annotated frames and N *persistent object representations*: object representation n refers to the same physical object in every frame, so temporal identity is given by construction and no tracking or Hungarian association is performed inside any graph model. The batch dimension of every forward pass is time ($B{=}T$), and all inputs are padded to $N_{\max}{=}64$ object representations and $K_{\max}$ person–object pairs. The inputs are an appearance feature $\mathbf{f}_n^t$ for every visible object, the oriented-box corners $\mathbf{C}_n^t \in \mathbb{R}^{8\times 3}$ of every valid object in the mode-dependent input frame described below, the validity mask $\mathrm{val}(n,t)$ (real object representation), the visibility mask $\mathrm{vis}(n,t)$ (object inside the camera view at t), the person–object pair indices $k=(i,j)$ with a pair-validity mask, the camera-to-world pose $\mathbf{T}_t \in \mathbb{R}^{4\times 4}$ with rotation $\mathbf{R}_t$ and translation $\boldsymbol{\tau}_t$ (from the π^3 annotation in every mode), a union-region appearance feature $\mathbf{u}_k^{t,\mathrm{roi}}$ for every pair, and, in PredCls only, the ground-truth object classes. As in the relationship triplets $(w_i^t, p_{ijk}^t, w_j^t)$ of Section 3, i indexes the subject and j the object of a pair; in ACTIONGENOME4D every annotated relationship has the person as its subject, so i is always the person. The appearance source is the only input that differs between methods (Unit 1 below). The outputs are, per frame, object-class logits for every object representation and three predicate distributions for every pair: *attention* (3 classes, softmax), *spatial* (6 labels, independent sigmoids) and *contacting* (17 labels, independent sigmoids). A pair whose object is not visible at t still receives a prediction; this is what makes the unobserved buckets of the evaluation scorable.

Coordinate convention. In PredCls, $\mathbf{C}_n^t = \mathbf{C}_n^{t,\mathrm{world}}$ comes from the world-frame annotation. In SGDet, the detector supplies camera-frame corners, $\mathbf{C}_n^t = \mathbf{C}_n^{t,\mathrm{cam}}$. These input corners are passed directly to the graph encoders; the camera-to-world transform is applied separately when comparing predicted boxes with world-frame targets or exporting world-frame boxes. Thus, the geometry formulas below describe the implemented computations on the supplied corners. Their interpretation as world-frame displacement or camera-relative geometry is valid for world-frame inputs; in SGDet, using camera-frame corners with world-frame poses does not provide that same geometric interpretation. Appendix F.3.2 distinguishes the input and comparison frames for corner refinement.

Four processing units. Every method is described, drawn and trained in the same four processing units: **Unit 1, Observed Objects Representation Processing Unit (O-ORPU)** (perception and the geometry scaffold of every object); **Unit 2, Unobserved Objects Representation Processing Unit (U-ORPU)** (how an object is represented while it is out of view); **Unit 3, Relationship Representation Processing Unit (RRPU)** (object context, person–object relationship representations and their temporal refinement); and **Unit 4, Prediction Heads** (the node and predicate heads and the losses on them).

Reading the method figures. Each method subsection has one figure that reads as one forward pass in the SGDet setting, left to right in a single row: the video, the frozen detector, Units 1 to 4 as tinted enclosures, and the method's output, the world scene graph. In PredCls, the annotated boxes and object classes take the place of the detector. Arrows that cross into the next unit name the tensor they carry, and long-range inputs run on lanes above or below the modules. Frozen modules are grey

Table 9: **Modules Of The Supervised WSGG Methods, By Processing Unit.** ✓ = present, – = absent; the value in brackets is the camera slice d_{camera} of the LKSTokenizer. Modules described in the method subsections rather than in Appendix F.1.2 are marked †.

Unit	Module	W-STTran	W-STTran++	W-DSGDetr	W-DSGDetr++	W-USG	WorldWise	WorldWise+	WorldWise++
1	ResNet-50 Faster R-CNN RoI / union features (frozen)	✓	✓	✓	✓	✓	–	–	–
1	Monocular 3D detector RoI / union features (frozen)	–	–	–	–	–	✓	–	–
1	DINOv3-L / π^3 Tier-1 tokens (frozen) + gated projector	–	–	–	–	–	–	✓	✓
1	Token grids + TokenGridFusion image memory†	–	–	–	–	–	–	–	✓
1	GlobalStructuralEncoder	✓	✓	✓	✓	✓	✓	✓	✓
1	ObjectSpatialEncoder	–	✓	✓	✓	–	✓	✓	✓
1	CameraTemporalEncoder (ego-motion)	–	–	–	–	–	✓	✓	✓
1	ObjectMotionEncoder	–	–	–	✓	–	✓	✓	✓
1	ScaffoldTokenizer (`[MASK]`, artificial masking)	–	–	–	–	–	✓	✓	✓
1+2	Entity decoder with free queries†	–	–	–	–	–	–	–	✓
2	LKS buffer (zero-order hold, non-differentiable)	✓	✓	✓	✓	✓	–	–	–
2	LKSTokenizer (d_{camera})	✓(0)	✓(128)	✓(128)	✓(128)	✓(0)	–	–	–
2	MotionFusion	–	–	–	✓	–	–	–	–
2	TemporalObjectEncoder	–	–	✓	✓	–	–	–	–
2	AssociativeRetriever	–	–	–	–	–	✓	✓	–
2	Visibility embedding + EMA reconstruction	–	–	–	–	–	✓	✓	✓
3	InterObjectTransformer (3D positional encoding)	✓	✓	✓	✓	–	✓	✓	–
3	ObjectContextEncoder (no 3D positional encoding)†	–	–	–	–	✓	–	–	–
3	Pair readout from decoder attention†	–	–	–	–	–	–	–	✓
3	Relationship representations + within-frame pair self-attention	✓	✓	✓	✓	✓	✓	✓	✓
3	with pair geometry $\boldsymbol{\gamma}_k^t$	–	–	–	–	–	✓	✓	✓
3	TemporalEdgeAttention	✓	✓	✓	✓	–	✓	✓	✓
3	USGRelationDecoder†	–	–	–	–	✓	–	–	–
4	NodePredictor, three predicate heads	✓	✓	✓	✓	✓	✓	✓	✓
4	CLIP alignment head†	–	–	–	–	✓	–	–	–
4	Box / corner refinement, free-query detection heads†	–	–	–	–	–	–	–	✓
–	Training loss (Appendix F.1.3)	LKS	LKS	LKS	LKS	LKS+$\mathcal{L}_{\text{align}}$	compl.	compl.	compl.+$\mathcal{L}_{\text{det}}$+$\mathcal{L}_{\text{refine}}$

with the frozen marker, learnable ones lilac, and modules new over the method they extend orange; a dashed box is non-differentiable. The $\mathcal{L}$ chips at the foot of a unit name the losses trained in it.

F.1.2 Shared Modules

The modules below are listed by the unit in which they sit. Table 9 gives, for every method, which of them it instantiates.

Unit 1: Observed Objects Representation Processing Unit (O-ORPU)

Appearance sources. Unit 1 receives, for every visible object, an appearance feature $\mathbf{f}_n^t$ and, for every person–object pair, a union-region feature $\mathbf{u}_k^{t,\text{roi}}$ of the same kind. Three frozen sources are used. (i) The five baselines use the 1024-d box-head output of a ResNet-50 Faster R-CNN. (ii) **WorldWise** uses the 1024-d box-head output of the monocular 3D detector of Appendix F.3.1, one detector per backbone (ResNet-50, DINOv2-B/L, DINOv3-L). (iii) WorldWise+ and WorldWise++ use Tier-1 tokens of two frozen foundation models. For these, both streams are cached once per video over every frame: the DINOv3 cacher runs the frozen `facebook/dinov3-vitl16-pretrain-lvd1689m` on the /16-padded 672 × 384 image and hooks layers L16, L20 and the final normalised layer L24n; the π^3 cacher runs the frozen `yyfz233/Pi3` on the 672 × 378 π^3 image, strips its five register tokens and hooks the frame-attention layers F4, F12–F16 and the global-attention layers G13–G15. Both use the same aspect-preserving resize (pixel limit 255,000), so a π^3 cell and a DINOv3 cell address the same pixels. Tier-1 tokens are obtained by `roi_align` (7×7, mean) per object box and per union box, for exactly the boxes of the detector feature files, giving the streams `dinov3_tok` = [L16 ‖ L20 ‖ L24n] (3,072-d), `pi3_tok` = [F4 ‖ F14 ‖ G14] (3,072-d) and `fused` (both, 6,144-d), written in the schema of the detector feature files. No gradient reaches any appearance source.

Gated fusion projector. In WorldWise+ and WorldWise++ the appearance projectors (the visual projector, d=256, and the union projector, d=64) are `GatedFusionProjectors` over the S concatenated streams. With $\mathbf{h}_s = \mathbf{W}_s\mathbf{f}_s$ the projection of stream s, a gate weighs each stream at each feature dimension:

$$\mathbf{g} = \text{softmax}_s\big(\mathbf{W}_g[\mathbf{h}_1\|\cdots\|\mathbf{h}_S]\big) \in \mathbb{R}^{S\times d}, \qquad \mathbf{z} = \text{LN}\Big(\text{ReLU}\Big(\textstyle\sum_s \mathbf{g}_s \odot \mathbf{h}_s\Big)\Big). \tag{31}$$

The single-stream case (S=1) reduces exactly to $\text{LN}(\text{ReLU}(\mathbf{W}\mathbf{f}))$, the projector of **WorldWise** with a 3,072-d input.

GlobalStructuralEncoder. Each oriented box is centred, $\bar{\mathbf{c}}_n^t = \frac{1}{8}\sum_{a=1}^{8}\mathbf{C}_{n,a}^t$, and the 24 centred corner coordinates are concatenated with the absolute centre, so the token keeps box shape, orientation and absolute position:

$$\mathbf{g}_n^t = \mathrm{MLP}_{\mathrm{obj}}\big([\mathrm{flatten}(\mathbf{C}_n^t - \bar{\mathbf{c}}_n^t) \,\|\, \bar{\mathbf{c}}_n^t]\big) \in \mathbb{R}^{256}, \tag{32}$$

with $\mathrm{MLP}_{\mathrm{obj}}$ of the form $27 \to 128 \to 256 \to \mathrm{LN} \to 256$ (ReLU). A max-pooled scene token is also computed; no method consumes it. The structural token exists for every valid object representation, visible or not.

ObjectSpatialEncoder. From the pose, the encoder computes the camera-to-centre vector $\mathbf{d}_n^t = \bar{\mathbf{c}}_n^t - \boldsymbol{\tau}_t$, its unit direction $\hat{\mathbf{d}}_n^t$, the viewing direction $\mathbf{v}_t = -\mathbf{R}_{t,:,2}/\|\mathbf{R}_{t,:,2}\|$ and the right axis $\mathbf{R}_{t,:,0}$, and encodes log-distance, view alignment and azimuth:

$$\mathbf{c}_n^t = \mathrm{MLP}_{\mathrm{cam}}\big([\log\|\mathbf{d}_n^t\|,\ \alpha_n^t,\ \hat{\mathbf{d}}_n^t \cdot \mathbf{R}_{t,:,0},\ \alpha_n^t]\big) \in \mathbb{R}^{128}, \qquad \alpha_n^t = \hat{\mathbf{d}}_n^t \cdot \mathbf{v}_t, \tag{33}$$

with $\mathrm{MLP}_{\mathrm{cam}}$ of the form $4 \to 64 \to 128$ (ReLU). It also produces a global camera token from the 6D rotation Zhou et al. (2019) and the translation, which no method consumes. Because geometry exists for out-of-view objects, $\mathbf{c}_n^t$ keeps changing during an occlusion.

CameraTemporalEncoder (ego-motion). For consecutive frames the relative pose $\mathbf{R}_{\mathrm{rel}}^{(t)} = \mathbf{R}_t\mathbf{R}_{t-1}^\top$, $\boldsymbol{\tau}_{\mathrm{rel}}^{(t)} = \boldsymbol{\tau}_t - \mathbf{R}_{\mathrm{rel}}^{(t)}\boldsymbol{\tau}_{t-1}$ (6D rotation and translation, 9 values) is encoded by an MLP ($9 \to 64 \to 64 \to \mathrm{LN} \to 128$; learned no-previous-frame embedding at $t{=}1$); a learned temporal position is added and a pre-norm encoder with 2 layers, 4 heads and feed-forward width 256 self-attends over the sequence, giving the ego-motion token $\mathbf{e}_t \in \mathbb{R}^{128}$.

ObjectMotionEncoder. From the supplied box centres the encoder computes a frame-to-frame displacement, gated so that a validity gap produces no spurious motion, its finite difference and a rotation-conditioned displacement,

$$\begin{aligned} \mathbf{v}_n^t &= \mathbb{1}[\mathrm{val}(n,t) \wedge \mathrm{val}(n,t{-}1)](\bar{\mathbf{c}}_n^t - \bar{\mathbf{c}}_n^{t-1}), \\ \boldsymbol{\mu}_n^t &= \mathrm{MLP}_{\mathrm{mot}}\big([\mathbf{v}_n^t \,\|\, \mathbf{a}_n^t \,\|\, \|\mathbf{v}_n^t\| \,\|\, \|\mathbf{R}_t^\top\mathbf{v}_n^t\| \,\|\, \|\mathbf{R}_t^\top\mathbf{v}_n^t\|]\big) \in \mathbb{R}^{64}, \end{aligned} \tag{34}$$

with $\mathrm{MLP}_{\mathrm{mot}}$ of the form $11 \to 32 \to \mathrm{LN} \to 32 \to 64$ and a learned no-motion embedding at $t{=}1$. The **WORLDWISE** family uses the gated acceleration $\mathbf{a}_n^t = \mathbf{v}_n^t - \mathbf{v}_n^{t-1}$; W-DSGDetr++ passes $\mathbf{a}_n^t = \mathbf{0}$ (velocity only). These are finite differences over represented frames and inherit the errors of box centres and poses.

ScaffoldTokenizer. The **WORLDWISE** family creates one object representation per object and represented frame. With a_n^t marking a visible cell masked artificially in training (probability $p_{\mathrm{mask}} = 0.3$ per visible valid cell, drawn afresh at every step) and the usable-observation indicator $u_n^t = \mathbb{1}[\mathrm{vis}(n,t) \wedge a_n^t{=}0]$, natural occlusions and artificial masks receive the same learned embedding:

$$\mathbf{z}_n^t = \begin{cases} \mathrm{VisProj}(\mathbf{f}_n^t), & u_n^t = 1, \\ \mathbf{e}_{[\mathrm{MASK}]}, & u_n^t = 0, \end{cases} \qquad \mathbf{x}_n^t = \mathrm{LN}\big(\mathrm{ReLU}\big(\mathbf{W}_{\mathrm{fuse}}[\mathbf{g}_n^t \,\|\, \mathbf{z}_n^t \,\|\, \mathbf{c}_n^t \,\|\, \boldsymbol{\mu}_n^t \,\|\, \mathbf{e}_t]\big)\big) \in \mathbb{R}^{256}, \tag{35}$$

where VisProj is $\mathrm{LN}(\mathrm{ReLU}(\mathbf{W}_v\mathbf{f}))$ ($1024 \to 256$) in **WORLDWISE** and the gated projector of Eq. (31) in WorldWise+ and WorldWise++, and the fusion input has $256 + 256 + 128 + 64 + 128 = 832$ dimensions. The tokenizer keeps a moving-average copy $\mathrm{VisProj}_{\bar{\theta}}$ of its projector, updated as $\bar{\theta} \leftarrow m\bar{\theta} + (1-m)\theta$ with $m = 0.996$ in every training forward pass; its output on the unmasked feature is the reconstruction target of Unit 2.

Unit 2: Unobserved Objects Representation Processing Unit (U-ORPU)

LKS buffer. The baselines' buffer is a bidirectional zero-order hold, computed without gradients. With t^- and t^+ the nearest visible frames of object n in the past and the future, the source frame is the nearer one (past on ties):

$$\mathbf{m}_n^t = \mathbf{f}_n^{t^*}, \qquad \Delta_n^t = |t - t^*|, \qquad t^* = \begin{cases} t^+, & (t^+ - t) < (t - t^-), \\ t^-, & \text{otherwise.} \end{cases} \tag{36}$$

A visible cell copies itself ($\Delta_n^t = 0$); a never-visible object representation receives $\mathbf{m}_n^t = \mathbf{0}$ and the sentinel $\Delta_n^t = 1000$. The buffer never synthesises appearance: an unobserved object is represented by its nearest observation and by how many frames ago it was made.

LKSTokenizer. Geometry, buffered appearance, the optional camera slice and the log-staleness are fused into one object representation,

$$\mathbf{x}_n^t = \mathrm{GELU}\big(\mathrm{LN}\big(\mathbf{W}_{\mathrm{fuse}}[\mathbf{g}_n^t \,\|\, \mathbf{m}_n^t \,\|\, \mathbf{c}_n^t \,\|\, \log(\Delta_n^t + 1)]\big)\big) \in \mathbb{R}^{256}, \tag{37}$$

with a $256 + 1024 + d_{\text{camera}} + 1$-dimensional input ($d_{\text{camera}} = 0$ drops $\mathbf{c}_n^t$). The raw 1024-d buffered feature enters the fusion layer directly, so the visual projection is learned inside $\mathbf{W}_{\mathrm{fuse}}$.

MotionFusion. A second projection injects the motion token into the object representation, $\hat{\mathbf{x}}_n^t = \mathrm{GELU}(\mathrm{LN}(\mathbf{W}_{\mathrm{mot}}[\mathbf{x}_n^t \,\|\, \boldsymbol{\mu}_n^t])) \in \mathbb{R}^{256}$, from a 320-d input.

TemporalObjectEncoder. The object representations are transposed to per-object sequences, a learned temporal position $\mathbf{P}_{\mathrm{obj}}(t)$ (300 positions) is added, and a pre-norm encoder with 2 layers, 4 heads, feed-forward width 512 and dropout 0.1 self-attends bidirectionally over the valid frames of each object:

$$\tilde{\mathbf{x}}_n^{1:T} = \mathrm{Enc}_{\mathrm{time}}\big(\{\mathbf{x}_n^t + \mathbf{P}_{\mathrm{obj}}(t)\}_{t=1}^{T}\big). \tag{38}$$

A frame in which the object representation holds a stale copy can thus read from all of its visible frames, before and after.

AssociativeRetriever. WORLDWISE and WorldWise+ complete every object representation's sequence from its own usable observations. With $\mathcal{V}_n = \{t : \mathrm{vis}(n,t) \wedge \mathrm{val}(n,t)\}$, the stacked object representations $\mathbf{X}_n \in \mathbb{R}^{T\times 256}$ and camera features $\mathbf{C}_n^{\mathrm{cam}}$, queries cover the whole sequence while keys and values come from $\mathcal{V}_n$ only:

$$\mathbf{Q}_n = \mathbf{X}_n + \mathrm{CamProj}_q(\mathbf{C}_n^{\mathrm{cam}}), \qquad \mathbf{K}_n = \mathbf{X}_{n,\mathcal{V}_n} + \mathrm{CamProj}_k(\mathbf{C}_{n,\mathcal{V}_n}^{\mathrm{cam}}), \qquad \mathbf{V}_n = \mathbf{X}_{n,\mathcal{V}_n}. \tag{39}$$

The retriever has 2 layers, each a multi-head cross-attention (4 heads, dropout 0.1) with a post-attention LayerNorm and a feed-forward block ($256 \to 512 \to 256$, GELU) with a second LayerNorm; after each layer the keys and values at source positions are refreshed with the updated queries. A naturally unobserved frame is never a source; an artificially masked frame stays in the pool with its `[MASK]` token, so the hidden feature itself is never available. If an object has no visible frame its first frame is unmasked as a fail-safe. Sources may lie before or after the queried frame, so the reported setting is offline world-state completion. The output is the completed sequence $\hat{\mathbf{X}}_n$.

Visibility embedding and EMA reconstruction. A learned two-state embedding $\mathbf{E}_{\mathrm{vis}} \in \mathbb{R}^{2\times 256}$, indexed by u_n^t, is added to the completed object representations, marking whether a representation rests on a usable observation or on completion; this gives the completed object representations $\hat{\mathbf{x}}_n^t$. A linear reconstruction head $\mathbf{W}_{\mathrm{rec}} \in \mathbb{R}^{256\times 256}$ maps $\hat{\mathbf{x}}_n^t$ back to the projected visual space, where it is compared with the EMA target $\mathrm{VisProj}_{\bar{\theta}}(\mathbf{f}_n^t)$ on the artificially masked cells (Eq. (43)). Naturally unobserved objects provide no reconstruction target.

Unit 3: Relationship Representation Processing Unit (RRPU)

InterObjectTransformer. Within each frame the object representations are shifted by a 3D spatial positional encoding and enriched by a pre-norm transformer encoder. For every ordered pair of valid objects the encoding takes the centre distance d_{ij}, the unit direction and the log-volume ratio (5 values), passes them through an MLP ($5 \to 64 \to 64$), averages over the valid neighbours and projects to 256:

$$\begin{aligned} \mathbf{p}_i^t &= \mathbf{W}_{\mathrm{pe}} \frac{1}{|\mathcal{N}_i|} \sum_{j\in\mathcal{N}_i} \mathrm{MLP}_{\mathrm{pair}}\big([d_{ij},\ (\bar{\mathbf{c}}_i - \bar{\mathbf{c}}_j)/d_{ij},\ \log V_i - \log V_j]\big), \\ \mathbf{h}^t &= \mathrm{Enc}_{\mathrm{obj}}(\{\mathbf{x}_n^t + \mathbf{p}_n^t\}_n), \end{aligned} \tag{40}$$

with 3 layers, 4 heads, feed-forward width 512, dropout 0.1 and a final LayerNorm; padded object representations are masked and zeroed. The outputs $\mathbf{h}_n^t \in \mathbb{R}^{256}$ are the enriched object representations.

Relationship representations and within-frame pair self-attention. For each valid pair $k = (i, j)$ the relationship representation concatenates the two enriched object representations, the projected union feature, the projected frozen CLIP embeddings $\mathbf{t}_c \in \mathbb{R}^{512}$ of the two classes and, in the **WORLDWISE** family, the projected pair geometry:

$$\begin{aligned}
\mathbf{u}_k^t &= \mathrm{LN}(\mathrm{ReLU}(\mathbf{W}_u \mathbf{u}_k^{t,\mathrm{roi}})) \in \mathbb{R}^{64}, \qquad \mathbf{e}_c = \mathbf{W}_{\mathrm{text}} \mathbf{t}_c \in \mathbb{R}^{128}, \\
\boldsymbol{\gamma}_k^t &= \mathrm{LN}(\mathrm{ReLU}(\mathbf{W}_\gamma \, \mathbf{q}_k^t)) \in \mathbb{R}^{64}, \\
\mathbf{q}_k^t &= \left[\hat{\mathbf{d}}_{ij},\ \log \|\mathbf{d}_{ij}\|,\ \log V_i,\ \log V_j,\ \log V_i - \log V_j,\ \log \min_{a,b} \|\mathbf{C}_{i,a} - \mathbf{C}_{j,b}\|\right] \in \mathbb{R}^8, \\
\mathbf{r}_k^t &= \mathrm{LN}\big(\mathrm{ReLU}(\mathbf{W}_{\mathrm{in}}[\mathbf{h}_i^t \,\|\, \mathbf{h}_j^t \,\|\, \mathbf{u}_k^t \,\|\, \mathbf{e}_{c(i)} \,\|\, \mathbf{e}_{c(j)} \,(\|\, \boldsymbol{\gamma}_k^t)])\big) \in \mathbb{R}^{256},
\end{aligned} \tag{41}$$

where $\mathbf{d}_{ij} = \bar{\mathbf{c}}_j^t - \bar{\mathbf{c}}_i^t$ points from the centre of the person i to that of the object j, and a, b index the eight corners of the two boxes. The input has 832 dimensions without pair geometry and 896 with it. The class $c(\cdot)$ is the ground truth in PredCls and the argmax of the node logits in SGDet; a learned no-union embedding replaces $\mathbf{u}_k^t$ when a batch has no union features. The K relationship representations of a frame then pass through a pre-norm encoder with 2 layers, 4 heads and feed-forward width 512.

TemporalEdgeAttention. The relationship representations are grouped by pair identity (i, j) across the video; a learned frame-position embedding is added and one pre-norm encoder layer (4 heads, feed-forward width 512) self-attends over each pair's sequence. The refined relationship representations are scattered back to their frames.

Unit 4: Prediction Heads

NodePredictor and predicate heads. The node head is an MLP $256 \to 256 \to 37$ (ReLU, dropout 0.1) giving logits for 36 foreground classes (person and 35 other categories) and background. In PredCls the classes are inputs and the logits are not used downstream; in SGDet they are trained and their argmax selects the class embedding of the relationship representations. Three MLP heads ($256 \to 128 \to C$) map every refined relationship representation to attention logits $\mathbf{a}_k^t \in \mathbb{R}^3$ (softmax), spatial logits $\mathbf{s}_k^t \in \mathbb{R}^6$ (sigmoid) and contacting logits $\mathbf{c}_k^t \in \mathbb{R}^{17}$ (sigmoid).

F.1.3 SHARED LOSSES

Let $\mathcal{P}$ be the valid person–object pairs over all frames of a video, $N_{\mathcal{P}} = |\mathcal{P}|$, $\mathcal{V} = \{k = (i, j) : \mathrm{vis}(j, t) = 1\}$ the pairs whose object is visible and $\mathcal{U} = \mathcal{P} \setminus \mathcal{V}$ those whose object is out of view (the person is always observed). For a pair k and targets $\mathbf{y}$, let $\ell_k(\mathbf{y}) = \mathrm{CE}(\mathbf{a}_k, y^{\mathrm{att}}) + \mathrm{BCE}(\mathbf{s}_k, \mathbf{y}^{\mathrm{spa}}) + \mathrm{BCE}(\mathbf{c}_k, \mathbf{y}^{\mathrm{con}})$, with a KL term to the target distribution in place of the cross-entropy when the attention target is soft. Every term below is a sum divided by the shared denominator $N_{\mathcal{P}}$, so each pair contributes to the gradient in proportion to its weight regardless of bucket size.

Bucketed LKS loss (baselines). Visible pairs use the manual labels $\mathbf{y}_k$ at full weight; pairs with an unobserved object use VLM pseudo-labels $\tilde{\mathbf{y}}_k$, smoothed with $\varepsilon = 0.2$ and down-weighted by $\lambda_{\mathrm{vlm}} = 0.2$:

$$\mathcal{L}_{\mathrm{LKS}} = \underbrace{\frac{1}{N_{\mathcal{P}}} \sum_{k \in \mathcal{V}} \ell_k(\mathbf{y}_k)}_{\mathcal{L}_{\mathrm{vis}}} + \underbrace{\frac{\lambda_{\mathrm{vlm}}}{N_{\mathcal{P}}} \sum_{k \in \mathcal{U}} \ell_k(\tilde{\mathbf{y}}_k)}_{\mathcal{L}_{\mathrm{vlm}}} + \mathbb{1}[\text{SGDet}]\, \mathcal{L}_{\mathrm{node}}. \tag{42}$$

The smoothed attention target puts $1 - \varepsilon + \varepsilon/3$ on the labelled class and $\varepsilon/3$ elsewhere; a multi-label target with k of C labels active puts $1 - \varepsilon$ on the active labels and $\varepsilon/(C - k)$ on the others. $\mathcal{L}_{\mathrm{node}}$ is the class cross-entropy of the node logits over valid object representations. No logit adjustment is applied.

Table 10: **Training Configuration Of The Supervised WSGG Methods** (`configs/methods/{predcls,sgdet}/`). Shared by all eight: AdamW, learning rate and weight decay 10^{-4}, 10% linear warm-up from $0.1\times$ then cosine annealing to $0.01\times$, gradient clipping 5.0, mixed precision, seed 0, one video per step ($B{=}T$), $N_{\max} = 64$, $d_{\text{model}} = d_{\text{struct}} = d_{\text{rel}} = 256$, $d_{\text{text}} = 128$, dropout 0.1, pair self-attention 2 layers / 4 heads / FF 512, best epoch by last-frame test with-constraint R@20 (no validation split). Layers / heads / FF are written $L{\times}H{\times}F$.

Parameter	W-STTran	W-STTran++	W-DSGDetr	W-DSGDetr++	W-USG	WORLDWISE	WorldWise+	WorldWise++
Appearance source	ResNet-50 Faster R-CNN box head					mono-3D detector box head	Tier-1 tokens	`dinov3_tok` + grids
d_{roi} / d_{union}	1024 / 1024					1024 / 1024	3072 (6144 fused)	3072 / readout
d_{camera} / d_{motion}	0 / –	128 / –	128 / –	128 / 64	0 / –	128 / 64		
Unit-2 temporal core	–	–	TOE $2{\times}4{\times}512$	TOE $2{\times}4{\times}512$	–	retriever $2{\times}4{\times}512$		decoder $4{\times}8{\times}1024$
Unit-3 object context	InterObjectTransformer $3{\times}4{\times}512$				context enc. $3{\times}4{\times}512$	InterObjectTransformer $3{\times}4{\times}512$		(decoder)
Temporal relation module	TemporalEdgeAttention $1{\times}4{\times}512$				USG decoder $2{\times}4{\times}512$	TemporalEdgeAttention $1{\times}4{\times}512$		
Pair geometry	off					on		
p_{mask}	–					0.3		
λ_{vlm} / label smoothing ε	0.2 / 0.2					0 / –		
$\lambda_{\text{recon}} \cdot \lambda_{\text{dom}}$, EMA m, τ	–					$0.5 \cdot 0.1$, 0.996, 0.5		
Method-specific terms	–				$\lambda_{\text{align}} = 0.1$, $\tau_{\text{align}} = 0.07$	–	–	Q=30, $\lambda_{\text{det}}{=}\lambda_{\text{refine}}{=}1$
Epochs (configured / completed)	20	20	20	20	20 / 15 (P), 11 (S)	20	20	20

Completion loss (WORLDWISE family). With $\mathcal{A} \subseteq \mathcal{V}$ the pairs whose object representation was artificially masked and $\mathcal{A}_{\text{cells}}$ the masked object–frame cells,

$$\mathcal{L}_{\text{compl}} = \mathcal{L}_{\text{SG}} + \mathcal{L}_{\text{sim}} + \lambda_{\text{recon}}\lambda_{\text{dom}}\,\mathcal{L}_{\text{recon}} + \mathbb{1}[\text{SGDet}]\,\mathcal{L}_{\text{node}}, \qquad \lambda_{\text{recon}} = 0.5,\ \lambda_{\text{dom}} = 0.1,$$

$$\mathcal{L}_{\text{SG}} = \frac{1}{N_{\mathcal{P}}} \sum_{k \in \mathcal{V}\setminus\mathcal{A}} \ell_k(\mathbf{y}_k), \qquad \mathcal{L}_{\text{sim}} = \frac{1}{N_{\mathcal{P}}} \sum_{k \in \mathcal{A}} \ell_k(\mathbf{y}_k),$$

$$\mathcal{L}_{\text{recon}} = \frac{1}{|\mathcal{A}_{\text{cells}}|} \sum_{(t,n) \in \mathcal{A}_{\text{cells}}} \left\| \mathbf{W}_{\text{rec}} \hat{\mathbf{x}}_n^t - \text{VisProj}_{\bar{\theta}}(\mathbf{f}_n^t) \right\|_2^2. \tag{43}$$

$\mathcal{L}_{\text{sim}}$ is the visible-pair loss evaluated on the artificially masked pairs with their clean labels, a partition of the visible bucket (the bucket follows the true visibility mask), not an extra weighted term. It supervises relationships of objects whose appearance was hidden, from their completed representations, without requiring a pseudo-label for a naturally unobserved pair. Naturally unobserved pairs receive no edge supervision ($\lambda_{\text{vlm}} = 0$) but are counted in $N_{\mathcal{P}}$; the effective reconstruction weight is 0.05. $\mathcal{L}_{\text{SG}}$ and $\mathcal{L}_{\text{sim}}$ are evaluated on logit-adjusted outputs Menon et al. (2021): with the training predicate priors π_c of the mode,

$$\Delta_c = \begin{cases} \tau \log \pi_c, & \text{attention}, \\ \tau \log \dfrac{\pi_c}{1-\pi_c}, & \text{spatial/contacting}, \end{cases} \qquad \tau = 0.5, \tag{44}$$

is added to the logits in training only; inference uses unadjusted logits. The baselines use VLM pseudo-labels at $\lambda_{\text{vlm}} = 0.2$ and no logit adjustment, so cross-family results compare complete systems, loss recipe included.

F.1.4 SHARED TRAINING PROCEDURE

Only the graph model is trained, from random initialisation; the appearance sources, the scaffold (corners and poses) and the CLIP class embeddings are frozen inputs. One model is trained per method, mode (PredCls, SGDet) and, where applicable, feature source on the 7,516 Train videos, one video per optimisation step ($B{=}T$ frames), with a configured 20-epoch schedule and AdamW (learning rate 10^{-4}, weight decay 10^{-4}), a linear warm-up from $0.1\times$ to the full learning rate over the first 10% of the steps followed by cosine annealing to $0.01\times$, gradient-norm clipping at 5.0, mixed precision and seed 0. In the **WORLDWISE** family a fresh artificial mask is drawn at every step and the EMA target is updated in every training forward pass. Checkpoint selection uses with-constraint R@20 on the last annotated frame of each test video after every epoch; there is no separate validation split. The adapted baselines and WorldWise selected on the earlier 1,526-video test annotation, while WorldWise+ and WorldWise++ selected on the 1,511-video test annotation used for reporting. Reported scores re-evaluate the selected checkpoints across all annotated frames. Because test annotations influenced epoch selection, these scores are subject to test-selection bias and are not estimates from a test set held out from model selection. Table 10 lists the per-method configuration; the method figures illustrate SGDet forward passes.

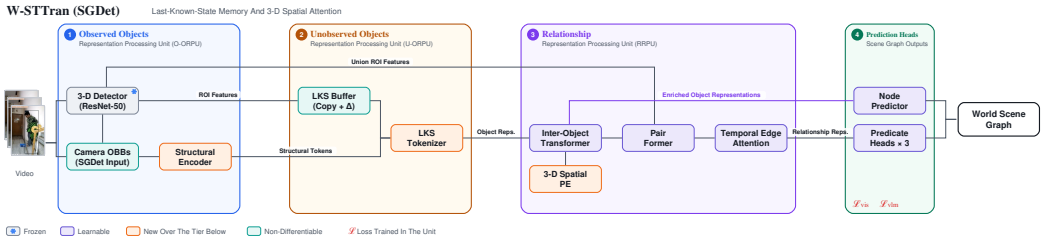


Figure 29: **Overview of the W-STTran Architecture.** We show the SGDet setting, in which the ResNet-50 variant of our frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes, lifted to the world frame with the camera poses, give the world-frame boxes of the objects, and its box head gives their appearance. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where the detector's box head gives the RoI appearance of each detected object and the union appearance of each person–object pair, and the GlobalStructuralEncoder turns the world-frame box of each object into a structural token that encodes its shape, orientation and position, whether or not the object is in view. In the U-ORPU, the non-differentiable LKS buffer fills every frame of an object with the RoI appearance of its nearest visible frame together with the staleness Δ of that copy, so an out-of-view object is represented by a copy of its closest observation; the LKSTokenizer then fuses the structural token, the copied appearance and the log-staleness into the object representation $\mathbf{x}_n^t$. Next, in the RRPU, the InterObjectTransformer enriches the object representations of each frame with scene context through a 3D positional encoding, the Pair Former fuses the representations of the person and the object with their union appearance and class embeddings into a relationship representation and applies within-frame attention among the pairs, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video, the only temporal step of the model. Finally, in the Prediction Heads, the node head classifies the enriched object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\text{vis}}$ with manual labels on pairs with a visible object, $\mathcal{L}_{\text{vlm}}$ with VLM pseudo-labels on pairs with an unobserved object). Orange modules are new over STTran.

F.2 Adapted Video Scene Graph Baselines

The four ladder baselines form a strict nested sequence of architectures, realised as separate model files rather than flags. W-STTran is the base: the LKS buffer and LKSTokenizer in Unit 2, the InterObjectTransformer, relationship representations and TemporalEdgeAttention in Unit 3, and the node and predicate heads in Unit 4. W-STTran++ adds the ObjectSpatialEncoder (Unit 1), W-DSGDetr adds the TemporalObjectEncoder (Unit 2), and W-DSGDetr++ adds the ObjectMotionEncoder (Unit 1) with MotionFusion (Unit 2). W-USG is not a rung of the ladder: it keeps the W-STTran substrate in Units 1 and 2 and replaces the Unit 3 relation stack by the components of USG-Par. All five read ResNet-50 Faster R-CNN features, have no ego-motion encoder, artificial masking, reconstruction objective or logit adjustment, and are trained with the bucketed LKS loss of Eq. (42) under the shared procedure of Appendix F.1.4; W-USG adds one alignment term. The modules are described in Appendix F.1.2; each subsection below gives the forward pass and the training objective of one baseline.

F.2.1 W-STTran

W-STTran adapts STTran Cong et al. (2021) to WSGG. It is the minimal configuration of the ladder: object persistence comes from a passive copy of the last observed features, spatial reasoning from one transformer over the objects of a frame, and temporal reasoning only from the relationship representations (Figure 29).

Forward pass. Unit 1: the frozen detector supplies the RoI feature $\mathbf{f}_n^t \in \mathbb{R}^{1024}$ of every visible object and the union feature of every pair; the GlobalStructuralEncoder turns the corners of every valid object into $\mathbf{g}_n^t$ (Eq. (32)). Unit 2: the LKS buffer holds, for every cell, the RoI feature of the nearest visible frame and its staleness Δ_n^t (Eq. (36)); the LKSTokenizer fuses $[\mathbf{g}_n^t \,\|\, \mathbf{m}_n^t \,\|\, \log(\Delta_n^t + 1)]$ (1,281-d, no camera slice) into the object representation $\mathbf{x}_n^t$ (Eq. (37)). Unit 3: the InterObjectTransformer enriches the object representations of each frame (Eq. (40)); the pair former builds the 832-d relationship representations without pair geometry and applies within-frame pair self-attention (Eq. (41)); TemporalEdgeAttention refines each pair across the video. This is the only temporal module of W-STTran; objects themselves are not encoded across time. Unit 4: the node

head reads the enriched object representations and the three predicate heads the refined relationship representations.

Training. W-STTran minimises the bucketed LKS loss $\mathcal{L}_{\text{vis}} + \mathcal{L}_{\text{vlm}} + \mathbb{1}[\text{SGDet}]\mathcal{L}_{\text{node}}$ of Eq. (42): manual labels at full weight on pairs with a visible object, VLM pseudo-labels smoothed with $\varepsilon = 0.2$ at weight $\lambda_{\text{vlm}} = 0.2$ on pairs with an unobserved object, both over the shared denominator $N_{\mathcal{P}}$. All loss terms sit on the Unit 4 heads; the buffer passes no gradient, so Units 1–3 learn only through the relation and node heads. Training follows Appendix F.1.4 (20 epochs, AdamW 10^{-4}, warm-up and cosine schedule).

F.2.2 W-STTRAN++

W-STTran++ is W-STTran with one addition in Unit 1: the ObjectSpatialEncoder, which tells every object representation how the current camera sees that object (Figure 30).

Forward pass. Unit 1: RoI and union features as in W-STTran, the structural token $\mathbf{g}_n^t$ (Eq. (32)), and the camera-relative feature $\mathbf{c}_n^t \in \mathbb{R}^{128}$ of every valid object (Eq. (33)). Unit 2: the LKS buffer (Eq. (36)) and the LKSTokenizer, now with the camera slice, $[\mathbf{g}_n^t \,\|\, \mathbf{m}_n^t \,\|\, \mathbf{c}_n^t \,\|\, \log(\Delta_n^t + 1)]$ (1,409-d, Eq. (37)). During an occlusion the copied appearance is constant, but $\mathbf{c}_n^t$ keeps following the camera. Unit 3 and Unit 4: unchanged from W-STTran (Eqs. (40) and (41), TemporalEdgeAttention, node and predicate heads).

Training. The bucketed LKS loss of Eq. (42) with $\lambda_{\text{vlm}} = 0.2$ and $\varepsilon = 0.2$, node cross-entropy in SGDet, no logit adjustment, masking or reconstruction, under the shared procedure of Appendix F.1.4.

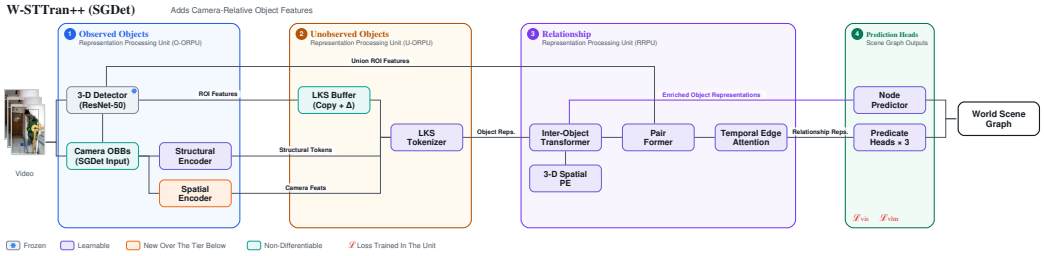


Figure 30: **Overview of the W-STTran++ Architecture.** W-STTran++ is W-STTran (Figure 29) with one addition in the O-ORPU. We show the SGDet setting, in which the ResNet-50 variant of our frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes, lifted to the world frame with the camera poses, give the world-frame boxes of the objects, and its box head gives their appearance. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where the detector's box head gives the RoI and union appearance, the GlobalStructuralEncoder encodes each world-frame box into a structural token, and the new ObjectSpatialEncoder encodes how the current camera sees each object (log-distance, view alignment and azimuth), a feature that keeps following the camera while the object is out of view. In the U-ORPU, the non-differentiable LKS buffer fills every frame of an object with the RoI appearance of its nearest visible frame together with the staleness Δ of that copy, so an out-of-view object is represented by a copy of its closest observation; the LKSTokenizer fuses the structural token, the camera-relative feature, the copied appearance and the log-staleness into the object representation $\mathbf{x}_n^t$, so a copied appearance now changes from frame to frame through the camera slice. Next, in the RRPU, the InterObjectTransformer enriches the object representations of each frame with scene context through a 3D positional encoding, the Pair Former fuses the representations of the person and the object with their union appearance and class embeddings into a relationship representation and applies within-frame attention among the pairs, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the enriched object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\text{vis}}$ with manual labels on pairs with a visible object, $\mathcal{L}_{\text{vlm}}$ with VLM pseudo-labels on pairs with an unobserved object).

F.2.3 W-DSGDETR

W-DSGDetr adapts DSGDetr Feng et al. (2022) to WSGG. DSGDetr encodes each tracked object across time before predicting relationships; the persistent object representations already provide the track, so the adaptation is a per-object temporal encoder at the end of Unit 2. W-DSGDetr contains every module of W-STTran++ and adds the TemporalObjectEncoder (Figure 31).

Forward pass. Unit 1: RoI and union features, structural tokens and camera-relative features, as in W-STTran++ (Eqs. (32) and (33)). Unit 2: the LKS buffer and the LKSTokenizer with the camera slice (Eqs. (36) and (37)), then the TemporalObjectEncoder (Eq. (38)): 2 pre-norm layers of bidirectional self-attention over the valid frames of each object, so a held copy is re-contextualised by all of the object's visible frames before and after. This is the differentiable temporal contextualisation that the zero-order hold alone does not provide; its output replaces the object representation for Unit 3. Unit 3 and Unit 4: unchanged from W-STTran.

Training. The bucketed LKS loss of Eq. (42) ($\lambda_{\text{vlm}} = 0.2$, $\varepsilon = 0.2$, node cross-entropy in SGDet) under the shared procedure of Appendix F.1.4; the TemporalObjectEncoder is trained end to end through the relation and node heads.

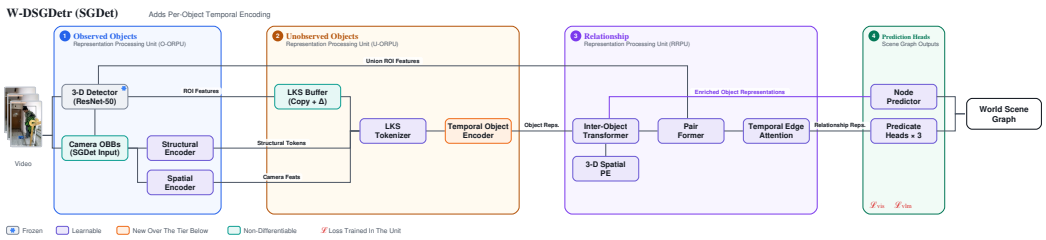


Figure 31: **Overview of the W-DSGDetr Architecture.** W-DSGDetr is W-STTran++ (Figure 30) with a per-object temporal encoder in the U-ORPU. We show the SGDet setting, in which the ResNet-50 variant of our frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes, lifted to the world frame with the camera poses, give the world-frame boxes of the objects, and its box head gives their appearance. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where the detector's box head gives the RoI and union appearance, and the GlobalStructuralEncoder and ObjectSpatialEncoder encode each world-frame box into a structural token and a camera-relative feature. In the U-ORPU, the non-differentiable LKS buffer fills every frame of an object with the RoI appearance of its nearest visible frame together with the staleness Δ of that copy, so an out-of-view object is represented by a copy of its closest observation; the LKSTokenizer fuses these features into the object representation $\mathbf{x}_n^t$, and the new TemporalObjectEncoder self-attends over all frames of each object, so that the copied frames are re-contextualised by the frames in which the object is visible, before and after the occlusion. Next, in the RRPU, the InterObjectTransformer enriches the object representations of each frame with scene context through a 3D positional encoding, the Pair Former fuses the representations of the person and the object with their union appearance and class embeddings into a relationship representation and applies within-frame attention among the pairs, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the enriched object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\text{vis}}$ with manual labels on pairs with a visible object, $\mathcal{L}_{\text{vlm}}$ with VLM pseudo-labels on pairs with an unobserved object).

F.2.4 W-DSGDETR++

W-DSGDetr++ is the most extensive baseline of the ladder. It contains every module of W-DSGDetr and adds the ObjectMotionEncoder in Unit 1 with a motion-fusion layer in Unit 2 that injects 3D object dynamics into the object representations before the temporal encoder (Figure 32). It has no ego-motion encoder, no masking and no reconstruction objective.

Forward pass. Unit 1: RoI and union features, structural tokens, camera-relative features, and the motion token $\boldsymbol{\mu}_n^t \in \mathbb{R}^{64}$ of the gated box-centre velocity (Eq. (34); the acceleration input is zero in this model, and frame 1 receives the learned no-motion embedding). Unit 2: the LKS buffer and the LKSTokenizer with the camera slice, then MotionFusion, $\hat{\mathbf{x}}_n^t = \mathrm{GELU}(\mathrm{LN}(\mathbf{W}_{\mathrm{mot}}[\mathbf{x}_n^t \,\|\, \boldsymbol{\mu}_n^t]))$ (320-d input), then the TemporalObjectEncoder (Eq. (38)). Unit 3 and Unit 4: unchanged from W-STTran.

Training. The bucketed LKS loss of Eq. (42) ($\lambda_{\mathrm{vlm}} = 0.2$, $\varepsilon = 0.2$, node cross-entropy in SGDet) under the shared procedure of Appendix F.1.4.

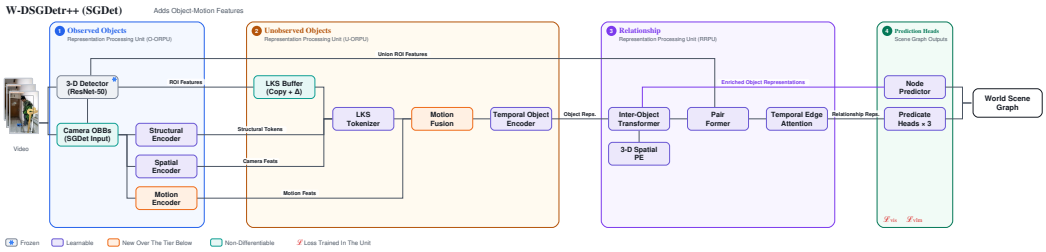


Figure 32: **Overview of the W-DSGDetr++ Architecture.** W-DSGDetr++ is W-DSGDetr (Figure 31) with the 3D motion of every object injected before the temporal encoder. We show the SGDet setting, in which the ResNet-50 variant of our frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes, lifted to the world frame with the camera poses, give the world-frame boxes of the objects, and its box head gives their appearance. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where the detector's box head gives the RoI and union appearance, the GlobalStructuralEncoder and ObjectSpatialEncoder encode each world-frame box into a structural token and a camera-relative feature, and the new ObjectMotionEncoder encodes the velocity of each box centre in the world and camera frames into a motion token. In the U-ORPU, the non-differentiable LKS buffer fills every frame of an object with the RoI appearance of its nearest visible frame together with the staleness Δ of that copy, so an out-of-view object is represented by a copy of its closest observation; the LKSTokenizer fuses these features into the object representation $\mathbf{x}_n^t$, the new MotionFusion adds the motion token to it, and the TemporalObjectEncoder re-contextualises the copied frames with the frames in which the object is visible. Next, in the RRPU, the InterObjectTransformer enriches the object representations of each frame with scene context through a 3D positional encoding, the Pair Former fuses the representations of the person and the object with their union appearance and class embeddings into a relationship representation and applies within-frame attention among the pairs, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the enriched object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\mathrm{vis}}$ with manual labels on pairs with a visible object, $\mathcal{L}_{\mathrm{vlm}}$ with VLM pseudo-labels on pairs with an unobserved object). The model has no ego-motion encoder, masking or reconstruction.

F.2.5 W-USG

W-USG transplants the relation machinery of USG-Par Wu et al. (2025c) onto the WSGG substrate. It is an external adaptation placed beside the ladder rather than on it: it keeps Units 1 and 2 of W-STTran, the node head and the construction of the relationship representations, and rebuilds Unit 3 with an object-context encoder and a relation decoder in which the relationship representations, used as the decoder's queries, cross-attend to the frame's context object representations; Unit 4 adds USG-Par's text-centric alignment head (Figure 33). The LKS buffer is kept because USG-Par is pixels-only and has no mechanism for emitting predictions on unobserved objects; without it the method would score zero on every unobserved pair by construction rather than by performance.

Forward pass. Unit 1 and Unit 2: as in W-STTran (RoI and union features, structural tokens, LKS buffer, LKSTokenizer without a camera slice); W-USG has no camera or motion encoder. Unit 3: the ObjectContextEncoder refines the object representations of a frame with a pre-norm encoder (3 layers, 4 heads, feed-forward width 512) *without* a 3D positional encoding, the analogue of USG-Par's 2D object-query refinement,

$$\mathbf{h}^t = \mathrm{Enc}_{\mathrm{ctx}}(\{\mathbf{x}_n^t\}_n) \in \mathbb{R}^{N\times 256}. \tag{45}$$

The pair former builds the relationship representations $\mathbf{r}_k^t$ of Eq. (41) (832-d, then 2 layers of within-frame pair self-attention), the relation proposals. The USGRelationDecoder, a pre-norm decoder with 2 layers, 4 heads and feed-forward width 512, alternates self-attention among the relationship representations of a frame and cross-attention from them to the frame's context object representations $\mathbf{h}^t$ (the memory),

$$\tilde{\mathbf{r}}^t = \mathrm{Dec}_{\mathrm{USG}}\left(\mathbf{r}^t;\ \mathbf{h}^t\right) \in \mathbb{R}^{K\times 256}. \tag{46}$$

There is no per-pair temporal attention: USG-Par predicts per frame after object association, and persistent object identity makes its associator the identity map, so persistence enters only through the buffer. Unit 4: the node and predicate heads, plus the alignment head, which projects every context object representation into the CLIP space and scores it against the frozen class embeddings,

$$z_{n,c}^t = \frac{1}{\tau_{\mathrm{align}}} \frac{\mathbf{W}_{\mathrm{al}}\mathbf{h}_n^t}{\|\mathbf{W}_{\mathrm{al}}\mathbf{h}_n^t\|} \cdot \frac{\mathbf{t}_c}{\|\mathbf{t}_c\|}, \qquad \tau_{\mathrm{align}} = 0.07,\ \ \mathbf{W}_{\mathrm{al}} \in \mathbb{R}^{512\times 256}. \tag{47}$$

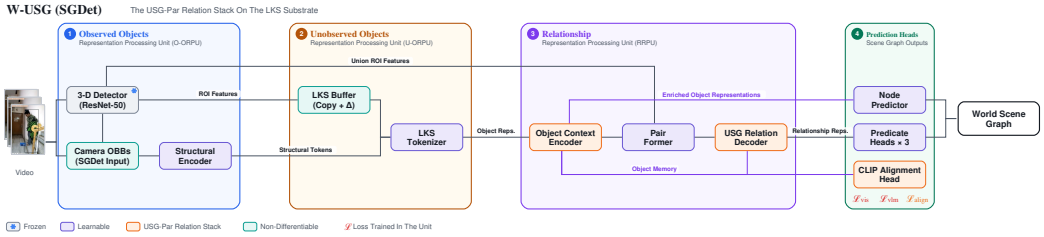


Figure 33: **Overview of the W-USG Architecture.** W-USG keeps the O-ORPU and U-ORPU of W-STTran (Figure 29) and replaces the RRPU by the relation stack of USG-Par (orange). We show the SGDet setting, in which the ResNet-50 variant of our frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes, lifted to the world frame with the camera poses, give the world-frame boxes of the objects, and its box head gives their appearance. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where the detector's box head gives the RoI and union appearance and the GlobalStructuralEncoder encodes each world-frame box into a structural token. In the U-ORPU, the non-differentiable LKS buffer fills every frame of an object with the RoI appearance of its nearest visible frame together with the staleness Δ of that copy, so an out-of-view object is represented by a copy of its closest observation; the LKSTokenizer fuses the structural token, the copied appearance and the log-staleness into the object representation $\mathbf{x}_n^t$, and this buffer is the only source of persistence in the model. Next, in the RRPU, the ObjectContextEncoder enriches the object representations of each frame with scene context *without* a 3D positional encoding, the Pair Former fuses the representations of the person and the object with their union appearance and class embeddings into relationship representations, and the USG Relation Decoder uses them as queries that alternate self-attention among the pairs of a frame with cross-attention to the frame's context object representations; there is no temporal attention across frames. Finally, in the Prediction Heads, the node head and the predicate heads produce the world scene graph ($\mathcal{L}_{\mathrm{vis}}$, $\mathcal{L}_{\mathrm{vlm}}$), and the CLIP alignment head scores every context object representation against the frozen class embeddings ($\mathcal{L}_{\mathrm{align}}$).

Relation-level text contrast is not included, because the repository ships CLIP embeddings for object classes only.

Training. `WUSGLoss` is the bucketed LKS loss with identical settings plus a cross-entropy of the alignment logits of every valid object–frame cell $(t, n) \in \mathcal{S}$ against its class,

$$\mathcal{L}_{\text{W-USG}} = \mathcal{L}_{\text{LKS}} + \lambda_{\text{align}} \underbrace{\frac{1}{|\mathcal{S}|} \sum_{(t,n)\in\mathcal{S}} \text{CE}\big(\mathbf{z}_n^t,\, c(n)\big)}_{\mathcal{L}_{\text{align}}}, \qquad \lambda_{\text{align}} = 0.1, \tag{48}$$

with $\mathcal{L}_{\text{LKS}}$ from Eq. (42). The alignment term applies in both modes, because the class labels are inputs in PredCls and targets in SGDet. Training follows Appendix F.1.4; the W-USG runs stopped after 15 of the 20 configured PredCls epochs and 11 of 20 SGDet epochs, and Table 2 uses the rescored checkpoints of these runs. The earlier frame-transfer experiment, which applied the Action-Genome-trained parser to each RGB frame and matched its predictions to world objects with ground-truth boxes, is not a row of the current comparison.

F.3 WORLDWISE METHODS

The **WORLDWISE** lineage consists of three graph models over the same persistent object representations as the baselines and one upstream detector. The monocular 3D detector supplies RoI features and, in SGDet, the 2D boxes and 3D corners of every object. **WORLDWISE** replaces the baselines' passive buffer by a differentiable masked-completion core and trains with the completion loss. WorldWise+ is **WORLDWISE** with its appearance input replaced by frozen foundation-model latents. WorldWise++ is WorldWise+ with its object-representation core replaced by an image-grounded entity decoder that also detects. The shared modules, the completion loss and the training procedure are described in Appendix F.1; the subsections below give the forward pass and the training objective of each graph model, with its figure in the four processing units in the SGDet setting.

F.3.1 MONOCULAR 3D DETECTOR

The monocular 3D object detector provides the geometric scaffolding for WSGG in the SGDet setting. Given a single RGB frame, it simultaneously produces 2D bounding boxes with class labels and 8-corner oriented 3D bounding boxes (OBBs) in camera coordinates. The graph encoders consume these camera-frame corners; extrinsic camera matrices transform them for comparison with world-frame targets and box export (Appendix F.1). The ResNet-50 W-baseline configurations are documented separately in Appendix F.2. Figure 34 shows its three stages. The system follows a Faster R-CNN Ren et al. (2016) architecture with three key components: (i) a **frozen DINOv2/v3 ViT backbone** Oquab et al. (2024) paired with a learnable Simple Feature Pyramid Network (SimpleFPN) Li et al. (2022b); (ii) a **factorized 3D prediction head** decomposing 3D box regression into dimensions, rotation, depth, and center offset, reconstructed via pinhole back-projection; (iii) an **OVMono3D-style disentangled 3D loss** Yao et al. (2025) with per-sample uncertainty weighting and Chamfer supervision.

Model Architecture. The detector first resizes the image and extracts frozen backbone features. A trainable feature pyramid supplies the region proposal network (RPN), and region-of-interest (RoI) heads predict object classes, 2D boxes, and 3D box parameters.

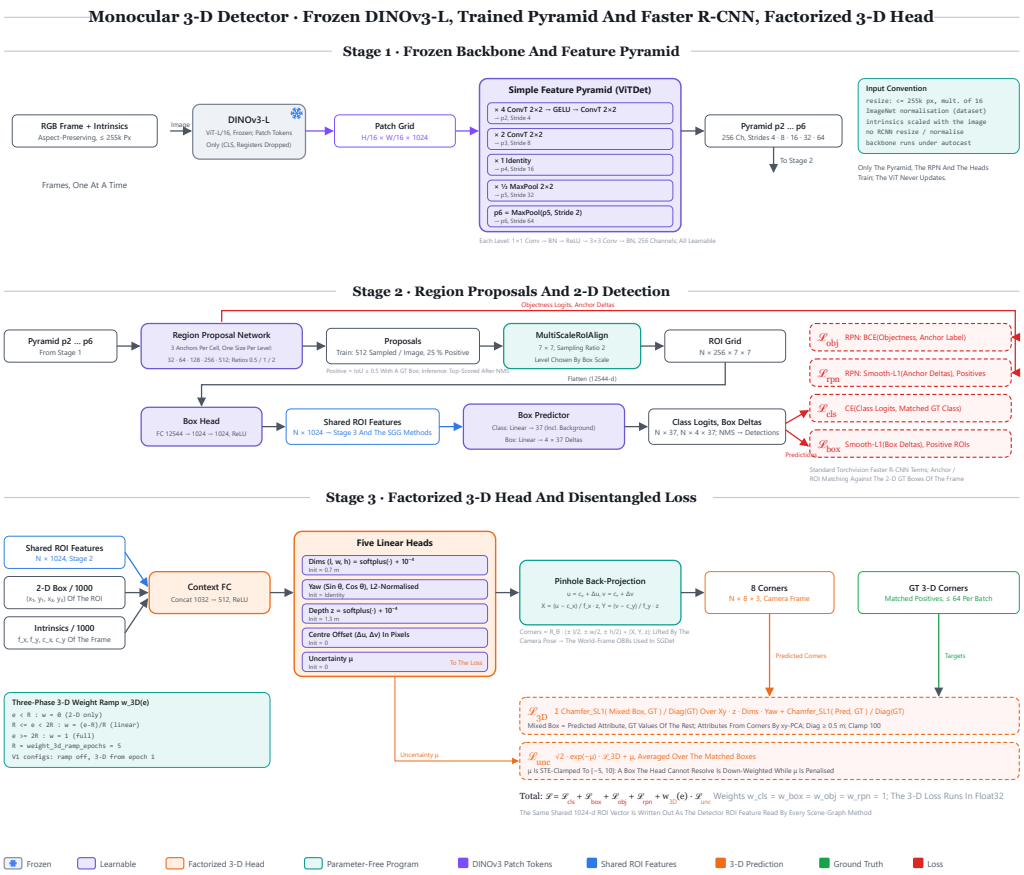


Figure 34: **Monocular 3D Detector: Backbone, Detection And Factorized 3-D Head.** One forward pass of the detector on a single RGB frame, in three stages. Stage 1 (backbone): the frame is resized with its aspect ratio preserved (pixel limit 255,000); the frozen DINOv3-L ViT turns it into patch tokens, from which the CLS and register tokens are stripped and the rest reshaped into a feature map; the learnable SimpleFPN upsamples it with transposed convolutions and downsamples it with max-pooling into the levels p2–p6, each projected to 256 channels, so that objects of every size are detected at a matching resolution. Stage 2 (2D detection): the region proposal network scores anchors of five sizes and three aspect ratios on every level and keeps the top proposals; MultiScaleRoIAlign pools each proposal from its level to 7×7, two fully connected layers give the shared 1024-d RoI feature, which is also the appearance feature that the **WORLDWISE** models read, and parallel heads predict the class scores and box refinements of the final 2D detections. Stage 3 (factorised 3D head): a separate branch concatenates the 1024-d RoI feature with the normalised 2D box and the camera intrinsics (1,032-d), passes it through a 512-d layer and predicts the dimensions (l, w, h), the yaw as $(\sin\theta, \cos\theta)$, the depth, a 2D centre offset and an uncertainty μ; the 3D centre is recovered by back-projecting the offset 2D centre at the predicted depth through the pinhole model, and the 8 corners are built from the dimensions, rotated by the yaw and translated to that centre. In SGDet, these camera-frame boxes are passed to the graph encoders; the camera-to-world transform is applied for world-frame box comparison and export. Training combines the Faster R-CNN classification, box and RPN losses with the OVMono3D disentangled 3D loss weighted by the predicted uncertainty, with the three-phase ramp of its weight.

Backbone: Frozen DINOv2/v3 ViT. The backbone is a Vision Transformer (ViT) Dosovitskiy et al. (2021) pretrained with DINOv2/v3 Oquab et al. (2024), kept entirely **frozen** during training. Four variants are supported (Table 11). Only the last $N_{\text{patch}} = H_p \times W_p$ spatial tokens are kept (stripping CLS and register tokens), reshaped to $\mathbf{F} \in \mathbb{R}^{B\times C\times H_p\times W_p}$.

Table 11: **Supported Monocular Vision Transformer Backbone Variants.**

Identifier	Model Backbone	Hidden Dim	Params	Patch Size
`v2`	DINOv2 ViT-B	768	86M	14
`v2s`	DINOv2 ViT-S	384	22M	14
`v2l`	DINOv2 ViT-L	1024	304M	14
`v3l`	DINOv3 ViT-L	1024	304M	16

Simple Feature Pyramid Network (SimpleFPN). Following ViTDet Li et al. (2022b), the SimpleFPN converts the single-scale backbone output into a multi-scale pyramid at strides $\{P/4, P/2, P, 2P, 4P\}$ via transposed convolutions (upsampling) and max-pooling (downsampling), where P is the patch size. Each level is projected to 256 channels by a 1×1 convolution, BatchNorm, ReLU, and a 3×3 refinement convolution. All FPN parameters are learnable.

Region Proposal Network and ROI Heads. The RPN uses standard Faster R-CNN anchors of sizes $\{32, 64, 128, 256, 512\}$ at pyramid levels $\{p2, \ldots, p6\}$ with aspect ratios $(0.5, 1.0, 2.0)$, contributing objectness and box-regression losses. Proposals are processed by MultiScaleRoIAlign (7×7 output, sampling ratio 2) feeding two FC layers that produce a shared 1024-d feature, from which parallel heads predict class logits and box deltas. This shared 1024-d feature is the RoI feature consumed by the graph models.

Factorized 3D Prediction Head. The 3D head is a lightweight branch sharing the 1024-d ROI features. It concatenates the shared features, normalized 2D box coordinates, and camera intrinsics ($1024 + 4 + 4 = 1032$ dimensions), then passes the result through a shared fully connected layer with ReLU activation and a 512-dimensional output. Five parallel heads produce:

$$\hat{\mathbf{d}} = \text{softplus}(\cdot) + \epsilon \in \mathbb{R}^3, \quad \text{(dimensions: } l, w, h\text{)} \tag{49}$$

$$[\hat{s}, \hat{c}] = \text{L2-normalize}(\cdot) \in \mathbb{R}^2, \quad \text{(rotation: } \sin\theta, \cos\theta\text{)} \tag{50}$$

$$\hat{z}_c = \text{softplus}(\cdot) + \epsilon \in \mathbb{R}^1, \quad \text{(depth)} \tag{51}$$

$$\hat{\delta} \in \mathbb{R}^2, \quad \text{(2D center offset)} \tag{52}$$

$$\hat{\mu} \in \mathbb{R}^1, \quad \text{(uncertainty)} \tag{53}$$

where $\epsilon = 10^{-4}$ keeps the predicted dimensions and depth strictly positive. The head is attached in one of two modes: *unified*, in which the 3D branch runs inside the RoI head on the shared features, or *separate*, in which it hooks the box head's output features and the 3D loss is computed outside the RoI heads. The reported detectors use the separate mode.

Pinhole Back-Projection. The 3D camera-frame center is recovered by intersecting the offset 2D center with the predicted depth via the pinhole model:

$$X_c = \frac{(u - c_x)}{f_x} \cdot \hat{z}_c, \quad Y_c = \frac{(v - c_y)}{f_y} \cdot \hat{z}_c, \quad Z_c = \hat{z}_c, \tag{54}$$

where $u = c_{x,\text{2d}} + \hat{\delta}_x$, $v = c_{y,\text{2d}} + \hat{\delta}_y$. The 8 corners are built at $(\pm l/2, \pm w/2, \pm h/2)$, rotated by the predicted yaw, and translated to $[X_c, Y_c, Z_c]^\top$.

Weight Initialization. Default initialization causes extreme variance in the initial 3D loss. We apply targeted initialization: Xavier-uniform for the context FC, $\mathcal{N}(0, 0.001)$ weights with biases chosen for depth (initial ≈1.3–1.8 m), dimensions (initial ≈0.7 m), and identity rotation ($\sin=0, \cos=1$).

OVMono3D Disentangled 3D Loss. The 3D loss follows OVMono3D Yao et al. (2025), combining uncertainty-weighted supervision with geometry-level disentanglement.

Disentangled Attribute Losses. Given predicted corners $\hat{\mathbf{C}}$ and GT corners $\mathbf{C}^* \in \mathbb{R}^{8\times3}$, both are decomposed via PCA into center $\mathbf{p}$, oriented dimensions $\mathbf{d} = (l, w, h)$, and yaw θ. For each attribute group $a \in \{xy, z, \text{dims}, r\}$, a "mixed" box uses the predicted attribute with GT values for all others:

Table 12: **Detector Training Configuration** of the reported checkpoints (`configs/detector/{dinov2,dinov2l,dinov3l}_*_separate_v1.yaml`, `resnet50_separate_v1.yaml`).

Configuration Key	DINO Detectors	ResNet-50	Description & Role
`model`	`v2`, `v2l`, `v3l`	`resnet50` (FPN-V2, fully fine-tuned)	Backbone variant (Table 11)
`lr`	10^{-4}	5×10^{-5}	Peak AdamW learning rate
`weight_decay`	10^{-3}	10^{-3}	AdamW decoupled weight decay
`batch_size`	128 (`v2`), 64 (`v2l`, `v3l`)	64	Per-GPU batch size
`epochs`	70	50	Total optimization epochs
`warmup_fraction`	0.01	0.01	Linear warm-up, then cosine annealing
`max_grad_norm`	1.0	1.0	Maximum gradient clipping threshold
`head_3d_mode`	`separate`	`separate`	3D head integration (unified or separate)
`head_3d_version`	`v1`	`v1`	3D head architectural formulation
`weight_3d`	1.0	1.0	Target weight for 3D loss objective
`weight_3d_ramp_epochs`	0	5	Three-phase loss ramp duration (R)
`max_3d_proposals`	64	64	Positive proposals per batch entering the 3D loss
`pixel_limit`	255,000	255,000	Maximum pixel count for aspect-preserving resize
`iou_match_2d_eval`	0.5	0.5	2D IoU for matching predictions to GT in evaluation

$$\mathcal{L}_{xy}: \quad \mathbf{C}_{xy} = \text{build}(\hat{p}_{xy} \| p_z^*, \mathbf{d}^*, \theta^*), \tag{55}$$

$$\mathcal{L}_{z}: \quad \mathbf{C}_{z} = \text{build}(p_{xy}^* \| \hat{p}_z, \mathbf{d}^*, \theta^*), \tag{56}$$

$$\mathcal{L}_{\text{dims}}: \quad \mathbf{C}_{d} = \text{build}(\mathbf{p}^*, \hat{\mathbf{d}}, \theta^*), \tag{57}$$

$$\mathcal{L}_{r}: \quad \mathbf{C}_{r} = \text{build}(\mathbf{p}^*, \mathbf{d}^*, \hat{\theta}). \tag{58}$$

An additional holistic loss $\mathcal{L}_{\text{all}} = \text{Chamfer}_{\text{SL1}}(\hat{\mathbf{C}}, \mathbf{C}^*)$ compares the full corners directly. All Chamfer computations use Smooth L1 distance for stability, and values are divided by the GT box diagonal for scale invariance.

Per-Sample Uncertainty Weighting. The uncertainty $\hat{\mu}_i$ is clamped with a straight-through estimator (STE) to $[-5, 10]$ and per-sample $\mathcal{L}_{\text{3D},i}$ is clamped at 100. The total per-sample 3D loss is:

$$\mathcal{L}_i = \sqrt{2} \cdot \exp(-\hat{\mu}_i) \cdot \mathcal{L}_{\text{3D},i} + \hat{\mu}_i, \quad \mathcal{L}_{\text{3D},i} = \mathcal{L}_{xy,i} + \mathcal{L}_{z,i} + \mathcal{L}_{\text{dims},i} + \mathcal{L}_{r,i} + \mathcal{L}_{\text{all},i}. \tag{59}$$

Total Training Loss. Let $\mathcal{I}_{\text{3D}}$ index the proposals with 3D supervision in the batch. The aggregate uncertainty-weighted loss is $\mathcal{L}_{\text{unc}} = |\mathcal{I}_{\text{3D}}|^{-1} \sum_{i \in \mathcal{I}_{\text{3D}}} \mathcal{L}_i$. The total loss combines all Faster R-CNN losses with this uncertainty-weighted 3D loss:

$$\mathcal{L}_{\text{total}} = w_{\text{cls}}\mathcal{L}_{\text{cls}} + w_{\text{box}}\mathcal{L}_{\text{box}} + w_{\text{obj}}\mathcal{L}_{\text{rpn-obj}} + w_{\text{rpn}}\mathcal{L}_{\text{rpn-box}} + w_{\text{3d}}(e) \cdot \mathcal{L}_{\text{unc}}, \tag{60}$$

where all weights are 1.0. A three-phase ramp is available for $w_{\text{3d}}(e)$: zero for the first R epochs (2D-only), linearly increasing from R to $2R$, and full weight thereafter. The trainer default is R=5; the V1 configurations used for the reported DINO checkpoints set R=0 and train the 3D loss from the first epoch, and the ResNet-50 configuration keeps R=5.

Dataset and Training. Data Sources. The data loader reads 2D annotations for 36 foreground classes (person and 35 other categories) from the ActionGenome files and 3D annotations from per-video files. The detector classifier adds background as its 37th output. The latter contain eight-corner OBBs in camera coordinates and camera intrinsics. Images are resized to an aspect-ratio-preserving resolution where both dimensions are multiples of patch size P, with total pixels $\leq$ 255K. Camera intrinsics are scaled proportionally. Images are grouped into *resolution buckets* sharing the same target size, enabling padding-free batching.

Training Configuration. Training uses AdamW Loshchilov & Hutter (2019) with linear warmup (1% of steps) followed by cosine annealing. Mixed precision is enabled via `GradScaler`, with gradients clipped at norm 1.0. Non-finite losses trigger batch skipping. CUDA optimizations include cuDNN auto-tuning, TF32 arithmetic, and Flash Attention. Key hyperparameters are listed in Table 12.

Evaluation. Evaluation computes 2D and 3D predictions in the same forward pass for each batch and aggregates the metrics over the test set.

2D Metrics. We report COCO-style mAP over IoU thresholds $[0.50, 0.55, \ldots, 0.95]$, along with mAP_{50} and mAP_{75}.

3D Metrics. 3D evaluation is performed on matched prediction–GT pairs (matched via 2D IoU $\geq$ 0.5, greedy, same-class). Box-level metrics include bidirectional Chamfer distance, corner L2, and oriented 3D IoU (via Sutherland–Hodgman polygon clipping Sutherland & Hodgman (1974)). Attribute-level metrics include center L2, dimensions L1, and wrapped rotation error. IoU hit rates at 50% and 75% measure 3D regression quality for correctly detected objects.

Table 13: **2D Detection Performance Across DINO Backbones** under separate and joint training modes. Bold indicates best per training mode.

Backbone	Training Mode	mAP	mAP_{50}	mAP_{75}
Separate Training (Independent 2D and 3D)				
DINOv2-B	Separate	0.0988	0.2333	0.0692
DINOv2-L	Separate	0.1103	0.2627	0.0739
DINOv3-L	Separate	**0.1762**	**0.3667**	**0.1461**
Joint Training (End-to-End with Staged Ramp)				
DINOv2-B	Joint	0.0998	0.2377	0.0672
DINOv2-L	Joint	0.1086	0.2610	0.0712
DINOv3-L	Joint	**0.1799**	**0.3660**	**0.1552**

Backbone and Training Mode Comparison. We evaluate the impact of backbone choice and training mode on 2D detection quality. Specifically, we compare three backbones: DINOv2-B (ViT-B/14, 86M), DINOv2-L (ViT-L/14, 304M), and DINOv3-L (ViT-L/16, 304M) under two training setups:

- **Separate**: The 2D detector and 3D head are trained independently; the 3D branch does not influence 2D detection gradients.
- **Joint**: Both 2D and 3D heads are trained end-to-end with the staged loss ramp (Eq. 60), allowing 3D supervision to refine shared representations.

Tables 13 and 14 distinguish 2D detection quality from 3D localization quality.

DINOv3-L has the highest 2D mAP. DINOv3-L achieves mAP of 0.18, a 60% relative improvement over DINOv2-L (0.11) and 80% over DINOv2-B (0.10). This gap is consistent across all IoU thresholds, with DINOv3-L reaching 0.37 mAP_{50} versus 0.26 for DINOv2-L. The comparison establishes a backbone-dependent detection difference; it does not separately identify the contribution of each pretraining or architectural change.

Table 14: **3D Localization Performance Across DINO Backbones** under separate and joint training. Mean IoU_{3D} measures average 3D volume overlap; Hit@0.50 denotes predictions with $\text{IoU}_{3D} \geq 0.50$.

Backbone	Training Mode	Mean IoU_{3D}	Hit@0.50
Separate Training (Independent 2D and 3D)			
DINOv2-B	Separate	**0.1446**	**0.0137**
DINOv2-L	Separate	0.1377	0.0117
DINOv3-L	Separate	0.1295	0.0111
Joint Training (End-to-End with Staged Ramp)			
DINOv2-B	Joint	**0.1411**	0.0120
DINOv2-L	Joint	0.1390	0.0104
DINOv3-L	Joint	0.1285	**0.0128**

Scaling from Base to Large within DINOv2 yields marginal gains. DINOv2-L improves over DINOv2-B by only $\sim$1 mAP point (0.11 vs. 0.10), in these detector configurations. The larger difference between DINOv2-L and DINOv3-L reflects the combined effect of their architecture and pretraining; this comparison does not isolate either factor.

Table 14 complements the 2D analysis with 3D localization quality.

3D localization ranking reverses the 2D ranking. Although DINOv3-L dominates 2D detection, DINOv2-B attains the best mean IoU_{3D} (0.1446) and Hit@0.50 under separate training. Under joint training, DINOv2-B still has the highest mean IoU, while DINOv3-L has the highest Hit@0.50. The downstream SGDet scores do not mirror this reversal: under the common ground truth of Appendix G.2, WorldWise with DINOv3-L exceeds its DINOv2-B counterpart in every all-relationship column (WC R@20 of 27.3 versus 19.5), in line with the larger share of visible objects its detector matches (60.5% versus 42.8%). Because SGDet matching tests only 2D boxes, these scores do not measure 3D localization quality. These cross-backbone results do not isolate localization error from other representation differences.

Numerical Stability Notes. All 3D loss computation is forced to float32 to avoid Chamfer overflow in float16. The 3D loss operates on at most 64 positive proposals per batch, randomly subsampled, to bound memory under the $5\times$ disentangled expansion. Resolution bucketing reads image headers for approximately 7,000 per-video dimension queries, avoiding full image decoding for size lookup.

F.3.2 WORLDWISE

WORLDWISE represents occlusion as missing visual evidence within a persistent object scaffold. An object keeps its representation when it leaves the image; Unit 2 completes that representation from the object's own visible observations, and Units 3 and 4 predict relationships from the completed representation (Figure 35). Against the baselines, it replaces the passive LKS copy by a differentiable masked-completion core and trains with the completion loss instead of pseudo-labels on unobserved pairs.

Forward pass. Unit 1: the frozen monocular 3D detector (one per backbone: ResNet-50, DINOv2-B/L, DINOv3-L) supplies the 1024-d box-head feature of every visible object and the union feature of every pair. Four encoders turn the scaffold into the structural token $\mathbf{g}_n^t$, the camera-relative feature $\mathbf{c}_n^t$, the ego-motion token $\mathbf{e}_t$ and the motion token $\boldsymbol{\mu}_n^t$ (Eqs. (32)–(34)), and the ScaffoldTokenizer fuses them with the projected appearance into one 256-d object representation per object and frame, substituting $\mathbf{e}_{[\mathrm{MASK}]}$ for every out-of-view cell and, in training, for 30% of the visible cells (Eq. (35)). Unit 2: the AssociativeRetriever (2 layers, 4 heads) completes the sequence of each object representation by cross-attention from every frame to the object's naturally visible frames only (Eq. (39)); the visibility embedding marks completed cells, and the linear reconstruction head maps the completed object representations back to the projected visual space. Unit 3: the InterObjectTransformer (Eq. (40)) lets the observed person interact with observed and completed objects; the pair encoder builds the 896-d relationship representations with pair geometry (Eq. (41)) and applies within-frame pair self-attention; TemporalEdgeAttention refines each pair across the video. Unit 4: the node head on the enriched object representations (used in SGDet) and the three predicate heads on the refined relationship representations. Retrieval sources may lie before or after

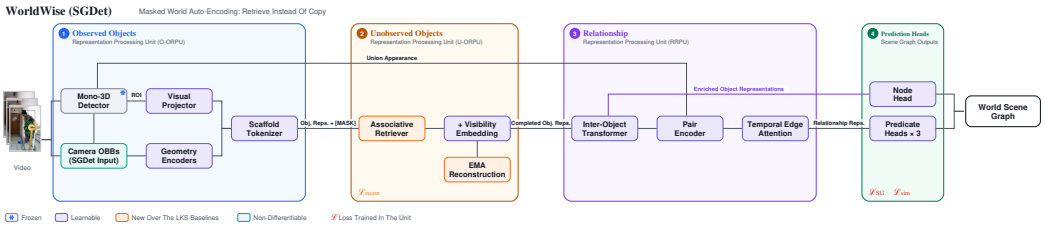


Figure 35: **Overview of the WorldWise Architecture.** We show the SGDet setting, in which the frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes supply camera-frame geometry to the graph encoders. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where the detector supplies the RoI appearance of each detected object and the union appearance of each person–object pair, and geometry encoders compute structural, spatial, object-motion and ego-motion features from the supplied corners and camera poses. The Scaffold Tokenizer fuses these features with the projected RoI appearance of the object, or with a learned `[MASK]` embedding when the object is out of view or, during training, artificially masked, thus forming one object representation $\mathbf{x}_n^t$ per frame for each object. Unlike the baselines, an out-of-view object is therefore not represented by a copy of old features but by a `[MASK]` embedding that the next unit completes. In the U-ORPU, the Associative Retriever lets every frame of an object attend to the frames in which that object is naturally visible, so that the masked frames are completed from the object's own observations; a visibility embedding is then added to distinguish observed objects from completed ones, giving the completed object representations $\hat{\mathbf{x}}_n^t$, and a reconstruction head predicts the hidden appearance of artificially masked objects against an exponential moving-average (EMA) target ($\mathcal{L}_{\mathrm{recon}}$). Next, in the RRPU, the InterObjectTransformer enriches the completed object representations of each frame with scene context through a 3D positional encoding, the Pair Encoder fuses the representations of the person and the object with their union appearance, class embeddings and pair geometry into a relationship representation and applies within-frame attention among the pairs, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the enriched object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\mathrm{SG}}$, $\mathcal{L}_{\mathrm{sim}}$, on logit-adjusted outputs).

the queried frame, so the reported setting is offline world-state completion; per-object temporal attention costs $O(MT^2d)$ for M objects and T frames, with a shorter source axis when only visible frames are used.

Training. **WORLDWISE** minimises the completion loss of Eq. (43),

$$\mathcal{L}_{\text{WorldWise}} = \mathcal{L}_{\text{SG}} + \mathcal{L}_{\text{sim}} + \lambda_{\text{recon}}\lambda_{\text{dom}}\,\mathcal{L}_{\text{recon}} + \mathbb{1}[\text{SGDet}]\,\mathcal{L}_{\text{node}}, \tag{61}$$

with $\lambda_{\text{recon}}\lambda_{\text{dom}} = 0.5 \cdot 0.1$. At every step a fresh artificial mask ($p_{\text{mask}} = 0.3$) hides the appearance of visible cells; $\mathcal{L}_{\text{sim}}$ supervises the relationships of those pairs with their clean manual labels from the completed object representations, $\mathcal{L}_{\text{recon}}$ pulls the reconstruction of the masked cells toward the EMA copy of the visual projector ($m = 0.996$), and $\mathcal{L}_{\text{SG}}$ supervises the unmasked visible pairs. Naturally unobserved pairs receive no edge supervision ($\lambda_{\text{vlm}} = 0$), and $\mathcal{L}_{\text{SG}}$ and $\mathcal{L}_{\text{sim}}$ use logit-adjusted outputs with $\tau = 0.5$ (Eq. (44)). One model is trained per mode and feature source under the shared procedure of Appendix F.1.4 (`worldwise_{predcls,sgdet}_{resnet50,dinov2b,dinov2l,dinov3l}.yaml`; the cells differ only in the feature source). Appendix G.5 reports the component ablation of this recipe.

F.3.3 WORLDWISE+

WorldWise+ is **WORLDWISE** with one seam re-plumbed, in Unit 1: the appearance input. Instead of the decoded box-head vector of the monocular detector, the model reads the latent tokens of two frozen foundation models, semantics from DINOv3-L and geometry from π^3 Wang et al. (2025c), RoI-pooled onto the very same object and union boxes. Nothing else changes, so WorldWise+ against **WORLDWISE** at DINOv3-L compares representations with the architecture held fixed (Figure 36).

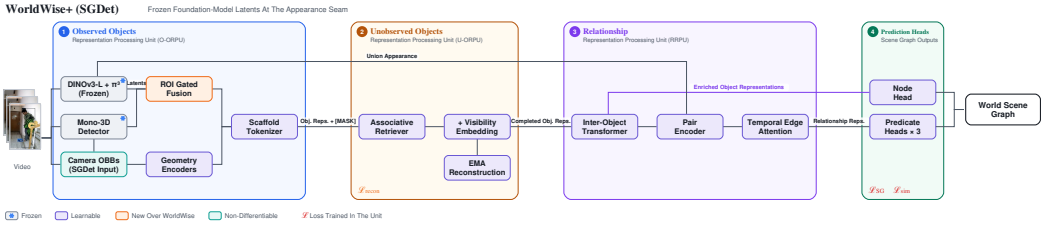


Figure 36: **Overview of the WorldWise+ Architecture.** WorldWise+ is **WORLDWISE** (Figure 35) with its appearance input replaced in the O-ORPU. We show the SGDet setting, in which the frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes supply camera-frame geometry to the graph encoders. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where frozen DINOv3-L semantic tokens and π^3 3D-aware tokens are RoI-pooled onto the detected box of each object and the union box of each person–object pair, and the new RoI gated fusion weighs the two streams at every feature dimension, replacing the visual and union projectors. In parallel, geometry encoders compute structural, spatial, object-motion and ego-motion features from the supplied corners and camera poses. The Scaffold Tokenizer fuses these features with the projected RoI appearance of the object, or with a learned `[MASK]` embedding when the object is out of view or, during training, artificially masked, thus forming one object representation $\mathbf{x}_n^t$ per frame for each object. In the U-ORPU, the Associative Retriever lets every frame of an object attend to the frames in which that object is naturally visible, so that the masked frames are completed from the object's own observations; a visibility embedding is then added to distinguish observed objects from completed ones, giving the completed object representations $\hat{\mathbf{x}}_n^t$, and a reconstruction head predicts the hidden appearance of artificially masked objects against an exponential moving-average (EMA) target ($\mathcal{L}_{\text{recon}}$). Next, in the RRPU, the InterObjectTransformer enriches the completed object representations of each frame with scene context through a 3D positional encoding, the Pair Encoder fuses the representations of the person and the object with their union appearance, class embeddings and pair geometry into a relationship representation and applies within-frame attention among the pairs, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the enriched object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\text{SG}}$, $\mathcal{L}_{\text{sim}}$, on logit-adjusted outputs).

Forward pass. Unit 1: the Tier-1 tokens of Appendix F.1.2 replace the detector features, $\mathbf{f}_n^t, \mathbf{u}_k^{t,\mathrm{roi}} \in \mathbb{R}^{3072}$ for the `dinov3_tok` and `pi3_tok` streams and $\mathbb{R}^{6144}$ for `fused`; the gated fusion projector (Eq. (31)) replaces both the visual projector of the ScaffoldTokenizer (d=256) and the union projector of the relationship representations (d=64), and its EMA copy becomes the reconstruction target. The geometry encoders and the tokenizer are those of **WORLDWISE** (Eqs. (32)–(35)). Unit 2, Unit 3 and Unit 4: unchanged from **WORLDWISE**: retriever, visibility embedding and reconstruction head; InterObjectTransformer, pair encoder with pair geometry and TemporalEdgeAttention; node and predicate heads.

Training. WorldWise+ minimises the same completion loss (Eq. (61)) with the reconstruction target $\mathrm{GatedProj}_{\bar{\theta}}(\mathbf{f}_n^t)$, the same $p_{\mathrm{mask}} = 0.3$, EMA momentum 0.996, $\lambda_{\mathrm{vlm}} = 0$ and logit adjustment with $\tau = 0.5$, under the shared procedure of Appendix F.1.4 with 4 data-loading workers. One model is trained per mode and stream (`worldwise_plus_<stream>_<mode>.yaml`); the stream cells differ only in the input width and the number of streams.

F.3.4 WORLDWISE++

WorldWise++ is the architectural step of the lineage. **WORLDWISE** and WorldWise+ never see an image: they reason over object representations whose appearance was pooled beforehand, and permanence lives in a separate module between the tokenizer and the inter-object transformer. WorldWise++ keeps the WorldWise+ front-end, the completion loss and the evaluation protocol, and replaces the object-representation core by one entity decoder, shared by Units 1 and 2, that cross-attends to the token grid of every frame, performs permanence and inter-object reasoning in one place, detects with free queries, and reads relations out of its own attention weights (Figure 37).

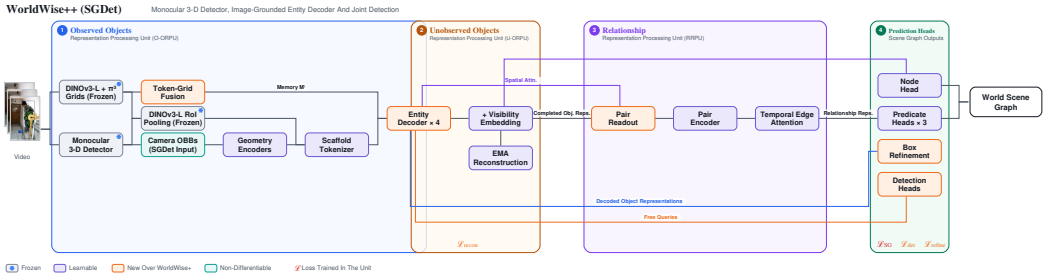


Figure 37: **Overview of the WorldWise++ Architecture.** We show the SGDet setting, as in Figure 3. The frozen monocular 3D detector (Appendix F.3.1) detects the objects of each frame; its 3D boxes supply camera-frame geometry to the graph encoders, and its 2D boxes select the region from which the DINOv3-L RoI appearance of each object is pooled. In PredCls, the annotated boxes and object classes take the place of the detector. The forward pass begins in the O-ORPU, where frozen DINOv3-L semantic features and π^3 3D-aware features of each frame are fused to construct the image memory $\mathbf{M}^t$, and geometry encoders compute structural, spatial, object-motion and ego-motion features from the supplied corners and camera poses. The Scaffold Tokenizer fuses these features with the projected DINOv3-L RoI appearance of the object, or with a learned `[MASK]` embedding when the object is out of view or, during training, artificially masked, thus forming one object-query $\mathbf{s}_n^t$ per frame for each object. Next, the Entity Decoder, which is shared by the O-ORPU and the U-ORPU, processes these object-queries together with 30 learned free queries: its temporal self-attention brings in information from the other frames of the same object, its spatial self-attention brings in context from the other objects in the frame, and its cross-attention to $\mathbf{M}^t$ grounds every object-query and free query in the whole image. It thereby constructs representations for both observed and unobserved objects, giving the decoded object representations $\hat{\mathbf{s}}_n^t$. In the U-ORPU, a visibility embedding is added to distinguish observed objects from constructed unobserved ones, giving the completed object representations $\hat{\mathbf{x}}_n^t$, and a reconstruction head predicts the hidden appearance of artificially masked objects against the EMA target ($\mathcal{L}_{\mathrm{recon}}$). Next, in the RRPU, the Pair Readout constructs a relationship representation for each person–object pair from the Entity Decoder's attention weights between the two objects and their decoded object representations, the Pair Encoder adds the class embeddings, the pair geometry and within-frame context, and Temporal Edge Attention refines each relationship representation with information from the other frames of the video. Finally, in the Prediction Heads, the node head classifies the completed object representations and the predicate heads classify the relationship representations to produce the world scene graph ($\mathcal{L}_{\mathrm{SG}}$, $\mathcal{L}_{\mathrm{sim}}$), the box-refinement heads correct the 2D box and 3D corners of each object from its decoded object representation ($\mathcal{L}_{\mathrm{refine}}$), and the detection heads decode the free queries into additional detections ($\mathcal{L}_{\mathrm{det}}$).

Forward pass. Unit 1: besides the `dinov3_tok` object features (the union tokens are ignored) and the scaffold, the model receives per frame two frozen token grids $\mathbf{G}^t_{\text{dino}}, \mathbf{G}^t_{\pi^3} \in \mathbb{R}^{H_p \times W_p \times 256}$: the DINOv3-L layer L24n, cropped to its unpadded region and resampled onto the π^3 lattice (27×48 cells for a 672×378 image), and the π^3 layer G14, each reduced to 256 channels by a frozen PCA fitted on training tokens (150 videos, about 500k tokens per stream). TokenGridFusion normalises and projects each stream and mixes them with a per-cell two-way gate,

$$\mathbf{w}^t_{xy} = \text{softmax}\big(\mathbf{W}_{\text{gate}}[\mathbf{a}^t_{xy} \,\|\, \mathbf{b}^t_{xy}]\big), \qquad \mathbf{M}^t_{xy} = \text{LN}\big(w^t_{xy,1}\mathbf{a}^t_{xy} + w^t_{xy,2}\mathbf{b}^t_{xy}\big) + \mathbf{P}_{\text{2D}}(x, y), \quad (62)$$

with (x, y) a grid cell, $\mathbf{a}^t = \mathbf{W}_{\text{dino}}\,\text{LN}(\mathbf{G}^t_{\text{dino}})$, $\mathbf{b}^t = \mathbf{W}_{\pi^3}\,\text{LN}(\mathbf{G}^t_{\pi^3})$ and a 2D sine encoding $\mathbf{P}_{\text{2D}}$, giving the image memory $\mathbf{M}^t$. The geometry encoders and the ScaffoldTokenizer (gated projector, $3072 \to 256$, `[MASK]` and artificial masking, EMA target) are those of WorldWise+; their outputs $\mathbf{s}^t_n$ are the persistent object-queries of the decoder.

Units 1 + 2, entity decoder: per frame the decoder processes $Q = 30$ learned free queries followed by the N object-queries, whose positional embedding is $\mathbf{p}^t_n = \mathbf{W}_{\text{pos}}\mathbf{g}^t_n$. Each of its $L = 4$ pre-norm layers ($d = 256$, 8 heads, feed-forward width 1,024, dropout 0.1) performs (i) *temporal self-attention* of every object-query over its valid frames with a learned frame-position embedding (bidirectional, not restricted to visible frames; this is where an unobserved or masked object-query reaches its own visible frames), (ii) *spatial self-attention* among all object-queries and free queries of a frame, whose per-head weights $\mathbf{A}^{t,\ell}$ are kept for the pair readout, (iii) *cross-attention* to the image memory $\mathbf{M}^t$, which grounds every object-query and free query in the whole frame beyond the pooled RoI and is what the free queries detect from, and (iv) a feed-forward block. A final LayerNorm yields the decoded free queries and the decoded object representations $\hat{\mathbf{s}}^t_n$. The decoder replaces the AssociativeRetriever and the InterObjectTransformer of **WorldWise**.

Unit 2: the visibility embedding is added to the decoded object representations, giving the completed object representations $\hat{\mathbf{x}}^t_n$, and the reconstruction head maps them to the projected visual space; the reconstruction now passes through the decoder.

Unit 3: for each pair $k = (i, j)$ the readout gathers the spatial attention weights in both directions from every layer and head and the element-wise product of the two decoded object representations, in the manner of EGTR Im et al. (2024),

$$\mathbf{u}^t_k = \text{LN}\big(\text{ReLU}(\mathbf{W}_{\text{pair}}[\alpha^t_k \,\|\, \hat{\mathbf{s}}^t_i \odot \hat{\mathbf{s}}^t_j])\big) \in \mathbb{R}^{64}, \qquad \alpha^t_k = [\mathbf{A}^{t,\ell}_h(i, j), \mathbf{A}^{t,\ell}_h(j, i)]_{\ell \le 4, h \le 8} \in \mathbb{R}^{64}, \quad (63)$$

which replaces the union projector; no union RoI features are used. The pair encoder (Eq. (41) on the completed object representations, with pair geometry), the within-frame pair self-attention and TemporalEdgeAttention follow as in **WorldWise**; the decoder's spatial self-attention takes the place of the InterObjectTransformer.

Unit 4: the node head and the three predicate heads as in **WorldWise**; two zero-initialised refinement heads ($256 \to 256 \to 4$ and $256 \to 256 \to 24$) on the decoded object representations, so that the untrained model returns the supplied geometry,

$$\hat{\mathbf{B}}^t_n = \mathbf{B}^t_n / [W, H, W, H] + \text{MLP}_{\text{box}}(\hat{\mathbf{s}}^t_n), \qquad \hat{\mathbf{C}}^{t,\text{in}}_n = \mathbf{C}^t_n + \text{MLP}_{\text{corner}}(\hat{\mathbf{s}}^t_n); \quad (64)$$

and detection heads on the free queries: a 38-way class distribution (the 37 node-head classes plus a separate no-object class at index 37), a normalised 2D box $\sigma(\text{MLP}_{\text{det}}(\mathbf{q}))$, and a camera-frame 3D box through the factorised parameterisation of the monocular detector (dimensions, yaw, depth and centre offset from the query and its box, corners by pinhole back-projection with $f = \max(H, W)$). The residual corners $\hat{\mathbf{C}}^{t,\text{in}}_n$ remain in the input frame: world coordinates in PredCls and detector camera coordinates in SGDet. Relationship readout uses the persistent object representations of the person and the objects; the free queries predict additional detections and are not substituted into the relationship evaluation.

Training. The objective extends the completion loss with a free-query detection term and a box-refinement term,

$$\mathcal{L}_{\text{WorldWise++}} = \underbrace{\mathcal{L}_{\text{SG}} + \mathcal{L}_{\text{sim}} + \lambda_{\text{recon}}\lambda_{\text{dom}}\mathcal{L}_{\text{recon}} + \mathbb{1}[\text{SGDet}]\,\mathcal{L}_{\text{node}}}_{\text{Eq. (43)}} + \lambda_{\text{det}}\,\mathcal{L}_{\text{det}} + \lambda_{\text{refine}}\,\mathcal{L}_{\text{refine}}, \qquad \lambda_{\text{det}} = \lambda_{\text{refine}} = 1. \quad (65)$$

The completion part is identical to **WORLDWISE** and WorldWise+ ($p_{\text{mask}} = 0.3$, EMA target, $\lambda_{\text{vlm}} = 0$, $\tau = 0.5$), so differences in recall are attributable to the architecture and the detection objective, not to the loss recipe. *Detection.* The targets of frame t are its valid, visible objects with a ground-truth 2D box. A Hungarian assignment between the Q free queries and the targets minimises $-p_{\text{cls}}(c_g) + 5\|\hat{\mathbf{b}}_q - \mathbf{b}_g\|_1 + \frac{1}{24}\|\hat{\mathbf{C}}_q - \mathbf{C}_g\|_1$ (predicted corners moved to the world frame with the pose); in training every unmatched query with box IoU above 0.5 to a target is also assigned to it (one-to-many, as in Hydra-SGG). With $\mathcal{M}$ the matches and $\mathcal{M}_{3\text{D}}$ those whose target has corners,

$$\mathcal{L}_{\text{det}} = \frac{1}{T}\sum_t \text{CE}_w\big(\hat{\mathbf{y}}^t_{\text{free}}, \text{tgt}_t\big) + \frac{5}{|\mathcal{M}|}\sum_{(q,g)\in\mathcal{M}} \|\hat{\mathbf{b}}_q - \mathbf{b}_g\|_1 + \frac{1}{|\mathcal{M}_{3\text{D}}|}\sum_{(q,g)\in\mathcal{M}_{3\text{D}}} \frac{\|\hat{\mathbf{C}}_q - \mathbf{C}_g\|_1}{24}, \tag{66}$$

where unmatched queries are assigned the no-object class at cross-entropy weight 0.1. *Box refinement.* Over the valid, visible objects with a ground-truth box,

$$\mathcal{L}_{\text{refine}} = \text{mean}\big\|\hat{\mathbf{B}}^t_n - \mathbf{B}^{t,\text{gt}}_n/[W,H,W,H]\big\|_1 + \text{mean}\big\|\hat{\mathbf{C}}^{t,\text{world}}_n - \mathbf{C}^{t,\text{gt}}_n\big\|_1, \tag{67}$$

where $\hat{\mathbf{C}}^{t,\text{world}}_n = \hat{\mathbf{C}}^{t,\text{in}}_n$ in PredCls. In SGDet, each refined corner is transformed by $\hat{\mathbf{c}}^{t,\text{world}}_{n,a} = \mathbf{R}_t\hat{\mathbf{c}}^{t,\text{in}}_{n,a} + \boldsymbol{\tau}_t$ before comparison with the world-frame target, as in the box-export code. All terms are trained jointly from the first step under the shared procedure of Appendix F.1.4 (at most 1,024 frames per video, 4 data-loading workers). The no-det ablation sets $Q = 0$ and $\lambda_{\text{det}} = \lambda_{\text{refine}} = 0$ while keeping the entity decoder.

G SUPERVISED WSGG EVALUATION AND WORLDWISE COMPONENT ABLATION

G.1 WORLDWISE AND WORLDWISE+: SUPPLEMENTARY COMPARISONS

The main paper describes WorldWise++ and its training paradigm, while Table 2 reports the adapted baselines and the DINOv3 configurations. Here we discuss every WorldWise and WorldWise+ configuration, as reported in Tables 15–19. Appendix F.3.2 describes WorldWise's associative masked-object completion, Appendix F.3.2 expands its encoders and supervision, and Appendix F.3.3 describes WorldWise+'s frozen token representations. WorldWise evaluates ResNet-50, DINOv2-B/L, and DINOv3-L features; WorldWise+ evaluates DINOv3, π^3, and fused token streams. These comparisons vary backbone, architecture, and supervision, and in SGDet also the detector, so they do not isolate the effect of each successive change.

PredCls comparison. In Table 15, with ResNet-50 on both sides, WorldWise reaches WC mR@20 of 49.1 versus 38.9 for W-DSGDetr++, although their supervision recipes also differ. WorldWise+ with DINOv3 tokens has the highest WC mR@20, 54.7, compared with 54.4 for WorldWise++, while WorldWise++ has the highest WC R@20 (74.8). Thus, the strongest variant depends on the metric. On UO, raw recall does not separate the model families: W-DSGDetr has the highest WC R@20 (61.6), and the WorldWise family spans 54.2–61.1. Mean recall does separate them, since every WorldWise-family configuration exceeds every adapted baseline in UO WC mR@20 (19.6–26.8 versus 13.7–16.8), and WorldWise++ is the highest.

SGDet comparison. In Table 17, WorldWise++ has the highest WC R@20 and mR@20 (29.5/16.7) and shares the highest NC mR@20 (30.9) with its no-det variant, while WorldWise+ with fused tokens has the highest NC R@20 (34.6). All six configurations that use the DINOv3 detector exceed every adapted baseline in every column of this table, for example in WC R@20 (27.3–29.5 versus 23.4–24.3). The ResNet-50 row of WorldWise uses the detector of the baselines and hence has the same object coverage, so it shows what the model adds on its own: its WC R@20 of 22.9 is below every baseline and its NC R@20 of 27.9 lies within their range (27.8–30.0), whereas its NC mR@20 of 24.2 exceeds all of them (16.3–20.5). With a shared detector, WorldWise therefore gains No-Constraint mean recall but not raw recall. Raw recall instead tracks the detector. In PredCls, where every configuration receives the same annotated objects, the four WorldWise backbones lie within 68.0–69.0 WC R@20; in SGDet they spread from 19.5 to 27.3, in the order of the share of visible objects that their detectors match (Appendix G.2). On UO (Tables 18 and 19),

every adapted baseline has a higher NC R@20 than every WorldWise-family configuration (20.0–22.1 versus 9.7–14.3). The gap also holds for the ResNet-50 row (10.6), so it does not come from detector coverage, and we have no ablation that isolates its cause. WorldWise++ has the highest UO NC mR@20 (11.8, versus 5.7–6.6 for the baselines).

G.2 Supervised WSGG operating points and visibility breakdown

Tables 15–19 extend the main results on the test split (1,511 videos). For each of the 14 supervised WSGG configurations and the adapted PUF rows (Appendix G.4), they report recall (R) and mean recall (mR) at $K = 10$ and $K = 20$, side by side in each row, under With Constraint (WC) and No Constraint (NC) evaluation. Results are separated into all relationships (All), relationships with observed objects (OO), and relationships with unobserved objects (UO). Because the subject is always the observed person, OO and UO partition the relationships. The tables are computed from the saved predictions of the reported runs and agree with Table 2 at the reported precision.

One ground truth for both settings. PredCls and SGDet are scored against the same annotated relationships: the PredCls ground truth of every test frame, 481,790 triplets, of which 414,443 involve an observed and 67,347 an unobserved object. It is identical for every method and backbone, and Tables 20 and 21 give its support per bucket and per predicate. In PredCls, the object representations are those of the annotated objects. In SGDet, every detected object representation used by a prediction is mapped to an annotated object of the same class. A visible object must in addition overlap the detected box at 2D IoU ≥ 0.5, whereas an unobserved object has no visible box and is matched by class, the identity its object representation carries. A triplet whose person or object receives no matching object representation counts as a miss, and a triplet counts once however many predictions recover it. Every R@K and mR@K is averaged over frames. R@K is the fraction of a frame's triplets recovered among its top K predictions, averaged over the frames that contain triplets of the bucket; mR@K first averages the per-frame recall of each predicate over the frames in which it occurs and then averages over all 26 predicates. Each bucket is averaged over its own frames, so the OO and UO values do not combine linearly into the All value. Because every method is scored against these annotated relationships, rather than against the labels attached to its own detected objects, the SGDet scores are comparable across detectors.

Detector coverage. A relationship can be recovered in SGDet only when its person and object both have a matching object representation, so the share of annotated objects that receive one bounds SGDet recall. Every detector matches the person in more than 95% of the frames. For the other objects, the ResNet-50 detector of the adapted baselines, WorldWise (ResNet-50), and the PUF rows matches 52.8% of the visible annotated objects and 46.1% of the unobserved ones. The DINOv2-B and DINOv2-L detectors match 42.8%/39.3% and 45.2%/40.4%, and the DINOv3 detector of WorldWise (DINOv3-L), WorldWise+, and WorldWise++ matches 60.5%/51.6%. This bound is one reason why SGDet recall stays far below PredCls recall for every configuration, and it makes SGDet differences between backbones partly a comparison of their detectors (Appendix G.1).

G.3 Qualitative supervised WSGG scene graphs

Figure 38 compares the PredCls scene graphs of W-DSGDetr++ and WorldWise++ on two test frames in which every object has a pre-extracted feature, so both models predict every edge. We show one frame on which WorldWise++ leads and one on which W-DSGDetr++ leads. For the unobserved door of `XACI3`, WorldWise++ predicts `not_looking_at`, `on_the_side_of`, and `not_contacting`, all of which are annotated, whereas W-DSGDetr++ predicts `in_front_of` as the spatial label. On `X37P1`, WorldWise++ predicts `looking_at` and `holding` for the unobserved clothes, neither of which is annotated, and W-DSGDetr++ recovers one more predicate over the frame.

G.4 Adapted persistent 3D-graph fusion

PUF Yang et al. (2026) is a training-free layer for online 3D scene-graph fusion: it associates observations with persistent nodes through a likelihood test, accumulates class and relation evidence as Dirichlet counts, and fills never co-observed edges with a class-conditional prior. We adapt it to test

Table 15: **Supervised WSGG PredCls With-Constraint (WC) performance across visibility buckets and K** on 1,511 test videos. Ground-truth localizations supplied. All metrics are percentages (%). Evaluates frame-averaged recall and predicate-averaged mR in every bucket. Each row reports R@K and mR@K at $K = 10, 20$ for every bucket. Rows under *PUF-Style Fusion* adapt PUF Yang et al. (2026) to the per-frame outputs of W-DSGDetr^{++} (Appendix G.4); they are not ranked, and † marks the non-causal bidirectional hold.

		All Relations				Observed Objects (OO)				Unobserved Objects (UO)			
Method	**Backbone / Stream**	**R@K**		**mR@K**		**R@K**		**mR@K**		**R@K**		**mR@K**	
		@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
Adapted Temporal Baselines *(LKS Memory Buffer)*													
W-STTran (Cong et al., 2021)	ResNet-50	62.5	68.2	30.9	37.6	64.7	69.9	32.4	39.1	46.9	58.7	12.5	14.8
W-STTran^{++} (Cong et al., 2021)	ResNet-50	61.3	66.9	28.0	34.0	63.0	68.0	29.1	35.1	48.5	59.5	14.2	16.8
W-DSGDetr (Feng et al., 2022)	ResNet-50	62.0	67.5	27.9	33.4	63.8	68.7	29.1	34.6	49.6	**61.6**	12.4	14.8
W-DSGDetr^{++} (Feng et al., 2022)	ResNet-50	62.8	68.5	32.1	38.9	65.2	70.3	33.9	40.7	46.3	58.2	11.4	13.7
W-USG (Wu et al., 2025c)	ResNet-50	61.6	67.3	27.5	33.4	63.6	68.7	28.7	34.6	47.6	58.7	13.3	15.7
WorldWise *(Iterative Masked Reconstruction)*													
WorldWise	ResNet-50	62.5	68.0	39.5	49.1	64.7	69.8	42.6	52.4	45.2	55.5	17.4	21.2
WorldWise	DINOv2-B	62.6	68.2	37.7	45.4	65.1	70.2	40.5	48.4	44.2	54.2	16.6	19.6
WorldWise	DINOv2-L	63.2	68.8	38.7	47.8	65.5	70.7	41.6	50.7	45.6	55.9	16.9	21.1
WorldWise	DINOv3-L	63.2	69.0	40.1	49.6	65.5	70.8	43.1	52.7	46.5	56.8	17.3	20.7
WorldWise+ *(Frozen Foundation Token Streams)*													
WorldWise+	DINOv3 tokens	67.6	73.5	**44.3**	**54.7**	70.5	76.0	**47.5**	**58.2**	46.9	56.9	19.8	24.1
WorldWise+	π^3 tokens	66.2	71.9	42.0	51.5	69.0	74.3	45.3	55.0	46.0	56.0	17.8	21.2
WorldWise+	Fused tokens	67.3	73.2	43.4	54.1	70.1	75.5	46.7	57.7	46.9	57.8	17.9	22.8
WorldWise++ *(Grounded Entity Decoder)*													
WorldWise++	DINOv3	**68.8**	**74.8**	43.9	54.4	**71.3**	**76.8**	47.1	57.8	**50.7**	61.1	**20.2**	**26.8**
WorldWise++	DINOv3 w/o det.	68.2	74.1	44.0	54.3	71.2	76.7	**47.5**	**58.2**	47.0	56.0	18.7	21.8
PUF-Style Fusion *(Yang et al., 2026; training-free, over W-DSGDetr^{++} per-frame outputs; not ranked)*													
LKS hold	causal	60.6	65.9	31.4	38.0	65.2	70.3	33.9	40.7	28.8	35.6	11.6	14.0
LKS hold	bidirectional†	62.6	68.1	32.3	39.2	65.2	70.3	33.9	40.7	43.8	54.2	15.8	19.6
PUF	$-$prior, $-$uncertainty	57.8	62.7	31.5	39.1	62.8	67.6	33.9	41.7	23.8	29.6	11.4	13.6
PUF	$-$prior	60.0	65.2	31.0	37.4	65.2	70.3	33.8	40.5	24.2	30.1	11.3	13.5
PUF	full	60.5	66.0	28.7	34.9	62.9	68.0	30.4	36.5	43.2	52.6	14.6	17.7
PUF	prior on UO pairs only	62.5	68.0	31.9	38.5	65.2	70.3	33.8	40.5	43.2	52.6	14.6	17.7

whether our unobserved-relationship gains follow from persistent memory alone. To this end, PUF fuses the per-frame outputs of the trained W-DSGDetr^{++} baseline on the same ResNet-50 detected objects; it uses π^3 points and camera poses in place of RGB-D, a new person node in every frame, and a prior fitted on the ACTIONGENOME4D training annotations. For each annotated frame, the graph built from the frames up to that frame is read out, so that observed pairs take the fused evidence of that frame, previously observed pairs take the evidence accumulated on their object node, and never co-observed pairs take the prior. We report the ladder in Tables 15–19: a last-known-state (LKS) hold, PUF without the prior and uncertainty (hard association with vote counting), PUF without the prior, full PUF, and full PUF with the prior restricted to unobserved pairs. The bidirectional LKS hold, marked †, looks up the nearest observed frame in either direction and is therefore non-causal. The PUF rows are scored by the same code and against the same ground truth as the supervised WSGG rows (Appendix G.2), and in SGDet they use the ResNet-50 detected objects of W-DSGDetr^{++}.

Insights: (1) Memory with a prior raises unobserved mean recall above its front end but not to the level of WorldWise on the same features. In PredCls, PUF with the prior on unobserved pairs reaches a UO NC mR@20 of 42.6, compared with 32.2 for its W-DSGDetr^{++} input, 44.6 for WorldWise on the same ResNet-50 features, and 51.3 for WorldWise++. Its UO NC R@20 of 82.5 is within 1.0 of the 83.5 of WorldWise++. The prior fills an unseen object with the frequent predicates of its class, which raises recall without recovering the rarer predicates that mean recall weights. (2) Applied to observed pairs, as in the original formulation, the same prior costs mean recall there: relative to PUF without the prior, full PUF lowers the OO WC mR@20 from 40.5 to 36.5 in PredCls and from 12.9 to 12.0 in SGDet. Restricting the prior to unobserved pairs avoids this cost, since its OO WC values equal those of PUF without the prior. (3) Uncertainty-aware association matters: in PredCls, hard association with vote counting has the lowest UO NC R@20 (42.2) and All NC R@20 (71.0) among the fusion arms, against 47.9 and 88.9 for PUF without the prior, which keeps the uncertainty-aware association. (4) In SGDet, memory adds little unobserved mean recall: the causal fusion rows reach between 5.0 and 7.4 UO NC mR@20 and the non-causal bidirectional hold reaches 8.0, against 6.6

Table 16: **Supervised WSGG PredCls No-Constraint (NC) performance across visibility buckets and K** on 1,511 test videos. Ground-truth localizations supplied. All metrics are percentages (%). Evaluates predicate recovery when multiple labels per object pair are permitted without ranking constraints. Each row reports R@K and mR@K at $K = 10, 20$ for every bucket. Rows under *PUF-Style Fusion* adapt PUF Yang et al. (2026) to the per-frame outputs of W-DSGDetr^{++} (Appendix G.4); they are not ranked, and † marks the non-causal bidirectional hold.

		All Relations				Observed Objects (OO)				Unobserved Objects (UO)			
Method	**Backbone / Stream**	**R@K**		**mR@K**		**R@K**		**mR@K**		**R@K**		**mR@K**	
		@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
Adapted Temporal Baselines *(LKS Memory Buffer)*													
W-STTran (Cong et al., 2021)	ResNet-50	74.6	92.6	44.8	70.0	77.3	94.0	47.5	72.9	53.0	80.4	13.4	36.4
W-STTran^{++} (Cong et al., 2021)	ResNet-50	73.8	92.4	41.0	63.6	75.6	93.3	43.1	66.1	56.0	82.9	16.2	34.5
W-DSGDetr (Feng et al., 2022)	ResNet-50	73.8	92.2	40.5	65.6	75.8	93.2	42.7	68.3	**57.2**	82.8	13.9	32.8
W-DSGDetr^{++} (Feng et al., 2022)	ResNet-50	74.7	92.7	47.0	71.0	77.5	94.3	50.3	74.4	53.3	79.9	12.4	32.2
W-USG (Wu et al., 2025c)	ResNet-50	74.0	92.6	39.8	64.0	75.9	93.5	42.0	66.4	55.1	**83.8**	14.7	34.6
WorldWise *(Iterative Masked Reconstruction)*													
WorldWise	ResNet-50	73.8	92.5	60.7	81.8	75.9	93.6	64.9	87.4	52.7	81.2	22.3	44.6
WorldWise	DINOv2-B	73.8	92.5	57.5	79.4	76.2	93.6	61.6	83.8	50.7	81.3	20.5	42.5
WorldWise	DINOv2-L	74.3	92.6	60.5	81.6	76.5	93.6	64.7	86.6	51.9	81.8	22.5	46.2
WorldWise	DINOv3-L	74.4	92.6	60.1	82.4	76.7	93.8	64.1	87.9	52.9	81.6	21.4	44.8
WorldWise+ *(Frozen Foundation Token Streams)*													
WorldWise+	DINOv3 tokens	77.7	94.3	65.5	85.6	80.6	95.6	70.2	91.4	52.2	82.5	25.8	48.5
WorldWise+	π^3 tokens	76.6	93.8	62.6	84.4	79.4	94.9	67.0	89.7	51.4	82.2	22.9	49.4
WorldWise+	Fused tokens	77.5	94.2	63.6	84.5	80.1	95.3	68.1	90.2	53.2	83.4	23.6	47.8
WorldWise++ *(Grounded Entity Decoder)*													
WorldWise++	DINOv3	**78.5**	**94.6**	63.8	**86.2**	80.9	**95.8**	68.4	91.5	56.5	83.5	**27.4**	**51.3**
WorldWise++	DINOv3 w/o det.	78.0	94.2	**65.6**	86.0	**81.0**	95.7	**70.5**	**92.0**	51.5	80.7	22.5	43.7
PUF-Style Fusion *(Yang et al., 2026; training-free, over W-DSGDetr^{++} per-frame outputs; not ranked)*													
LKS hold	causal	72.7	89.2	47.5	70.0	78.3	94.6	51.4	75.3	32.8	50.8	13.5	23.8
LKS hold	bidirectional†	74.4	92.9	47.4	70.8	76.7	93.9	49.6	73.9	53.1	82.8	20.2	36.0
PUF	$-$prior, $-$uncertainty	62.8	71.0	38.8	49.3	68.1	75.4	42.3	52.8	25.7	42.2	11.6	20.3
PUF	$-$prior	72.1	88.9	46.7	69.4	78.3	94.5	50.9	74.7	28.5	47.9	13.6	27.4
PUF	full	72.5	91.1	39.5	64.5	74.8	92.4	41.6	66.9	51.0	79.7	18.6	38.5
PUF	prior on UO pairs only	74.2	92.8	46.4	70.5	76.8	93.8	49.3	73.2	50.3	82.5	18.4	42.6

for the W-DSGDetr^{++} input and 11.8 for WorldWise++. The PUF rows are not ranked, and none of them exceeds the best ranked value in any column. In PredCls, where every row receives the same annotated objects, WorldWise (ResNet-50) and WorldWise++ exceed every PUF row in UO WC and NC mR@20, so we attribute their unobserved-relationship gains to learned completion rather than to persistence alone. In SGDet, the same holds for WorldWise++, whose detector also covers more unobserved objects, whereas WorldWise on the shared ResNet-50 detector is on par with the best causal PUF row (UO NC mR@20 of 7.6 versus 7.4). We note that this adaptation departs from PUF in its geometry source, scale normalization, and relation heads, and that its fitted 3D boxes are weak (mean oriented IoU of 0.15 on observed PredCls objects), so we do not report a 3D-IoU comparison.

G.5 WorldWise component ablation study

Table 22 ablates the components of WorldWise with DINOv3-L features in PredCls: each row removes one component from the full model, adds VLM pseudo-label supervision, or raises the logit-adjustment temperature. The study ablates WorldWise, not WorldWise+ or WorldWise++. It is not scored on the 1,511-video test split of Table 2, so its rows are compared with one another and not with the main-table scores. Its SGDet counterpart is not reported, because it was not scored against the common ground truth of Appendix G.2.

Removing logit adjustment raises raw recall and lowers mean recall. Adding VLM supervision lowers all eight metrics. Removing pair geometry lowers WC mR@10 from 40.11 to 37.61. These patterns support a trade-off analysis of the components of WorldWise.

The EMA-target row changes target construction while retaining the reconstruction loss. Removing artificial masking weakens a training signal while retaining retrieval. Neither experiment is a direct removal of the reconstruction objective, AssociativeRetriever, or visibility embedding. Direct re-

Table 17: **Overall supervised WSGG SGDet results** on 1,511 videos, with and without the ranking constraint. Every method is scored against the same annotated relationships as PredCls (481,790 triplets), and recall is averaged over frames. A detected object must match an annotated object of the same class at 2D IoU ≥ 0.50; an unobserved object is matched by class, the identity its object representation carries. A relationship is recalled when both of its objects are matched and the predicate agrees; relationships whose objects are not matched count as misses. Values are percentages. Each row reports R@K and mR@K at $K = 10, 20$ under both constraints. Rows under *PUF-Style Fusion* adapt PUF Yang et al. (2026) to the per-frame outputs of W-DSGDetr^{++} (Appendix G.4); they are not ranked, and † marks the non-causal bidirectional hold.

		With Constraint (WC)				**No Constraint (NC)**			
Method	**Backbone / Stream**	**R@K**		**mR@K**		**R@K**		**mR@K**	
		@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
Adapted Temporal Baselines *(LKS Memory Buffer)*									
W-STTran (Cong et al., 2021)	ResNet-50	12.6	23.6	4.7	12.2	15.6	28.3	8.9	18.1
W-STTran^{++} (Cong et al., 2021)	ResNet-50	12.4	23.4	4.4	11.2	14.7	27.8	8.0	16.4
W-DSGDetr (Feng et al., 2022)	ResNet-50	13.0	24.2	4.8	12.2	16.9	29.5	9.2	18.3
W-DSGDetr^{++} (Feng et al., 2022)	ResNet-50	13.3	24.3	4.8	12.8	17.1	30.0	10.9	20.5
W-USG (Wu et al., 2025c)	ResNet-50	12.7	23.8	4.7	12.4	15.8	28.0	7.7	16.3
WorldWise *(Iterative Masked Reconstruction)*									
WorldWise	ResNet-50	12.2	22.9	4.7	12.9	16.4	27.9	14.9	24.2
WorldWise	DINOv2-B	11.2	19.5	5.2	12.7	14.7	23.2	13.9	21.0
WorldWise	DINOv2-L	11.9	20.5	5.4	12.3	15.8	24.2	14.4	20.8
WorldWise	DINOv3-L	15.0	27.3	5.9	14.5	20.2	32.4	16.5	26.5
WorldWise+ *(Frozen Foundation Token Streams)*									
WorldWise+	DINOv3 tokens	16.2	28.9	6.3	15.7	<u>21.3</u>	33.9	**19.7**	30.5
WorldWise+	π^3 tokens	15.8	28.5	6.2	15.5	20.7	33.5	17.8	28.6
WorldWise+	Fused tokens	<u>16.3</u>	29.2	<u>6.4</u>	16.2	**21.7**	**34.6**	<u>19.6</u>	<u>30.8</u>
WorldWise++ *(Grounded Entity Decoder)*									
WorldWise++	DINOv3	**16.5**	**29.5**	**6.6**	**16.7**	21.1	<u>34.5</u>	19.3	**30.9**
WorldWise++	DINOv3 w/o det.	**16.5**	<u>29.4</u>	<u>6.4</u>	<u>16.4</u>	21.1	<u>34.5</u>	**19.7**	**30.9**
PUF-Style Fusion *(Yang et al., 2026; training-free, over W-DSGDetr^{++} per-frame outputs; not ranked)*									
LKS hold	causal	12.8	23.7	4.7	12.7	16.9	29.6	11.0	20.9
LKS hold	bidirectional†	12.9	24.0	4.7	12.8	16.6	29.5	10.8	20.5
PUF	$-$prior, $-$uncertainty	12.1	20.9	4.3	11.1	20.0	27.0	16.9	20.7
PUF	$-$prior	12.2	22.9	4.5	11.9	15.5	27.8	9.3	18.4
PUF	full	13.2	24.1	4.5	11.1	18.9	28.5	9.3	15.5
PUF	prior on UO pairs only	12.3	23.1	4.5	12.0	15.6	27.9	9.3	18.3

movals of those components have not been evaluated here. Measuring their individual contributions would require matched ablations scored with the annotations and evaluator of Table 2.

Table 18: **Supervised WSGG SGDet visibility buckets, With Constraint (WC)** across 1,511 test videos. Every method is scored against the same annotated relationships as PredCls (481,790 triplets: 414,443 OO and 67,347 UO). A detected object must match an annotated object of the same class at 2D IoU ≥ 0.50; an unobserved object is matched by class, the identity its object representation carries. A relationship is recalled when both of its objects are matched and the predicate agrees; relationships whose objects are not matched count as misses. Recall is averaged over frames, and mR@K additionally over predicates. All values are percentages (%). Each row reports R@K and mR@K at $K = 10, 20$ for both buckets. Rows under *PUF-Style Fusion* adapt PUF Yang et al. (2026) to the per-frame outputs of W-DSGDetr^{++} (Appendix G.4); they are not ranked, and † marks the non-causal bidirectional hold.

		Observed Objects (OO)				**Unobserved Objects (UO)**			
Method	**Backbone / Stream**	**R@K**		**mR@K**		**R@K**		**mR@K**	
		@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
Adapted Temporal Baselines *(LKS Memory Buffer)*									
W-STTran (Cong et al., 2021)	ResNet-50	13.1	25.0	5.0	13.1	9.5	15.3	2.1	4.4
W-STTran^{++} (Cong et al., 2021)	ResNet-50	12.8	24.6	4.6	11.8	**9.6**	**15.8**	2.1	4.5
W-DSGDetr (Feng et al., 2022)	ResNet-50	13.6	25.5	5.0	13.0	9.5	15.3	2.1	4.4
W-DSGDetr^{++} (Feng et al., 2022)	ResNet-50	13.9	25.7	5.1	13.7	**9.6**	15.3	2.1	4.4
W-USG (Wu et al., 2025c)	ResNet-50	13.2	25.1	4.9	13.2	9.4	15.4	2.2	4.5
WorldWise *(Iterative Masked Reconstruction)*									
WorldWise	ResNet-50	13.3	24.7	5.1	13.8	4.8	11.0	2.5	5.9
WorldWise	DINOv2-B	12.2	21.1	5.7	13.7	4.8	9.9	2.4	5.2
WorldWise	DINOv2-L	12.9	22.1	5.9	13.5	4.9	10.1	2.3	4.7
WorldWise	DINOv3-L	16.3	29.5	6.4	15.7	6.3	13.4	2.9	6.8
WorldWise+ *(Frozen Foundation Token Streams)*									
WorldWise+	DINOv3 tokens	17.6	31.2	6.9	16.9	6.6	14.0	**3.2**	8.3
WorldWise+	π^3 tokens	17.3	30.7	6.8	16.7	6.7	14.0	2.9	7.6
WorldWise+	Fused tokens	17.8	31.5	7.0	17.5	6.8	14.3	**3.2**	8.4
WorldWise++ *(Grounded Entity Decoder)*									
WorldWise++	DINOv3	**18.0**	**31.7**	**7.2**	**18.0**	7.3	15.0	**3.2**	**9.4**
WorldWise++	DINOv3 w/o det.	**18.0**	31.6	7.0	17.7	7.4	15.0	**3.2**	9.1
PUF-Style Fusion *(Yang et al., 2026; training-free, over W-DSGDetr^{++} per-frame outputs; not ranked)*									
LKS hold	causal	13.9	25.7	5.1	13.7	5.2	10.0	2.4	5.2
LKS hold	bidirectional†	13.9	25.7	5.1	13.7	5.9	12.2	2.4	5.8
PUF	$-$prior, $-$uncertainty	13.3	22.8	4.7	11.8	4.6	8.2	2.3	4.1
PUF	$-$prior	13.4	24.9	4.9	12.9	4.4	9.3	2.4	5.1
PUF	full	14.2	26.0	5.0	12.0	6.0	11.6	2.3	5.1
PUF	prior on UO pairs only	13.4	24.9	4.9	12.9	5.4	11.0	2.4	5.3

Table 19: **Supervised WSGG SGDet visibility buckets, No Constraint (NC)** across 1,511 test videos. Every method is scored against the same annotated relationships as PredCls (481,790 triplets: 414,443 OO and 67,347 UO), without ranking constraints. A detected object must match an annotated object of the same class at 2D IoU ≥ 0.50; an unobserved object is matched by class, the identity its object representation carries. A relationship is recalled when both of its objects are matched and the predicate agrees; relationships whose objects are not matched count as misses. Recall is averaged over frames, and mR@K additionally over predicates. All values are percentages (%). Each row reports R@K and mR@K at $K = 10, 20$ for both buckets. Rows under *PUF-Style Fusion* adapt PUF Yang et al. (2026) to the per-frame outputs of W-DSGDetr^{++} (Appendix G.4); they are not ranked, and † marks the non-causal bidirectional hold.

		Observed Objects (OO)				**Unobserved Objects (UO)**			
Method	**Backbone / Stream**	**R@K**		**mR@K**		**R@K**		**mR@K**	
		@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
Adapted Temporal Baselines *(LKS Memory Buffer)*									
W-STTran (Cong et al., 2021)	ResNet-50	16.2	29.3	9.3	18.9	**11.3**	21.4	2.7	6.0
W-STTran^{++} (Cong et al., 2021)	ResNet-50	15.2	28.6	8.3	17.0	**11.3**	**22.1**	2.5	6.2
W-DSGDetr (Feng et al., 2022)	ResNet-50	18.0	30.8	9.7	19.2	9.5	20.0	2.4	5.8
W-DSGDetr^{++} (Feng et al., 2022)	ResNet-50	18.4	31.5	11.6	21.7	8.6	20.0	2.3	6.6
W-USG (Wu et al., 2025c)	ResNet-50	16.6	29.0	8.1	17.1	9.5	20.2	2.2	5.7
WorldWise *(Iterative Masked Reconstruction)*									
WorldWise	ResNet-50	18.2	30.5	16.0	25.9	4.1	10.6	3.7	7.6
WorldWise	DINOv2-B	16.4	25.3	15.2	22.8	4.7	9.7	3.0	8.5
WorldWise	DINOv2-L	17.6	26.4	15.8	22.7	4.7	9.9	3.1	5.8
WorldWise	DINOv3-L	22.6	35.4	18.0	28.8	5.2	12.8	3.3	7.7
WorldWise+ *(Frozen Foundation Token Streams)*									
WorldWise+	DINOv3 tokens	24.0	37.3	**21.7**	33.2	5.1	12.1	3.2	10.1
WorldWise+	π^3 tokens	23.1	36.7	19.4	31.1	5.7	12.8	4.3	8.5
WorldWise+	Fused tokens	**24.3**	**37.8**	21.4	**33.5**	5.6	13.3	3.5	9.3
WorldWise++ *(Grounded Entity Decoder)*									
WorldWise++	DINOv3	23.5	37.7	21.1	33.4	6.1	14.1	**4.6**	**11.8**
WorldWise++	DINOv3 w/o det.	23.5	37.6	21.6	33.4	6.0	14.3	4.4	11.5
PUF-Style Fusion *(Yang et al., 2026; training-free, over W-DSGDetr^{++} per-frame outputs; not ranked)*									
LKS hold	causal	18.8	32.3	11.9	22.4	4.6	11.5	2.7	6.8
LKS hold	bidirectional†	18.2	31.7	11.6	21.8	5.6	14.2	3.0	8.0
PUF	$-$prior, $-$uncertainty	22.4	29.5	18.2	22.0	4.7	9.7	3.8	6.7
PUF	$-$prior	17.2	30.3	10.0	19.7	4.1	11.0	2.5	6.6
PUF	full	20.7	30.9	10.0	16.7	6.5	11.7	2.8	5.0
PUF	prior on UO pairs only	17.2	30.0	10.0	19.4	4.8	13.9	2.4	7.4

Table 20: **Ground-truth support across visibility buckets in the 1,511-video test evaluation.** Evaluates total relation triplets (subject, object, predicate, timestamp) across 48,834 annotated frames. Supervised WSGG SGDet is scored on the PredCls ground truth, so it has the same support as PredCls for every detector backbone. Because the person subject is always observed, every triplet belongs to the OO or the UO bucket.

Variant	**Protocol**	**Evaluation GT Layout**	**Total Triples**	**OO**	**UO**
Supervised* WSGG *Methods					
Supervised	PredCls	Curated objects with pre-extracted features	481,790	414,443	67,347
Supervised	SGDet	PredCls ground truth, every detector backbone	481,790	414,443	67,347
Zero-Shot* WSGG *And U*-WSGG *Methods (Full Inventory)					
Zero-shot	PredCls	Unlocalized: complete annotated inventory	773,236	414,443	358,793
Zero-shot	PredCls	Localized: complete annotated inventory	773,236	414,443	358,793
Zero-shot	SGDet	Localized: complete annotated inventory	773,236	414,443	358,793

Table 21: **Supervised WSGG per-predicate ground-truth support in OO and UO visibility buckets** across 1,511 test videos. SGDet is scored on the PredCls ground truth, so both settings share this support.

Predicate	PredCls and SGDet	
	OO	**UO**
Attention Axis ($\lvert\mathcal{P}_{\text{att}}\rvert = 3$)		
looking_at	47,620	3,400
not_looking_at	67,328	16,282
unsure	11,287	230
Spatial Axis ($\lvert\mathcal{P}_{\text{spa}}\rvert = 6$)		
above	2,063	526
beneath	21,441	742
in_front_of	83,159	9,195
behind	16,596	3,977
on_the_side_of	24,240	8,971
in	4,295	1,446
Contacting Axis ($\lvert\mathcal{P}_{\text{con}}\rvert = 17$)		
carrying	1,433	924
covered_by	1,790	45
drinking_from	1,215	42
eating	913	11
have_it_on_the_back	147	37
holding	50,265	1,621
leaning_on	3,708	41
lying_on	1,149	23
not_contacting	35,586	16,577
other_relationship	3,363	57
sitting_on	14,540	42
standing_on	2,842	11
touching	16,995	2,333
twisting	27	99
wearing	2,012	694
wiping	192	13
writing_on	237	8
Total Triplets	**414,443**	**67,347**

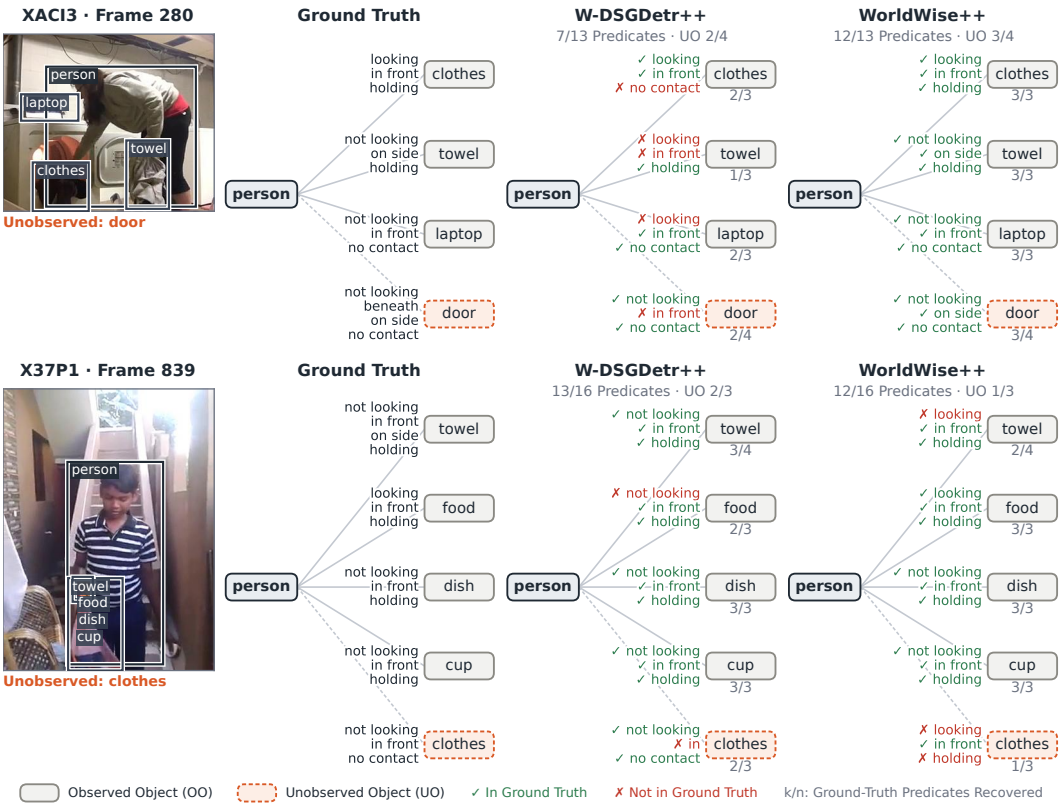


Figure 38: **Supervised WSGG PredCls scene graphs on two test frames.** Left to right: the frame with its annotated boxes, the ground-truth graph, and the graphs predicted by W-DSGDetr++ and WorldWise++, each with one attention, one spatial, and one contacting predicate per edge (With Constraint). ✓ marks a predicted predicate that is in the ground truth of its edge and ✗ one that is not; k/n under a node counts the recovered ground-truth predicates of that edge, and the panel headers sum them over the frame and over its unobserved objects. On these frames every prediction lies within the top 20 of the frame, so the marks equal the hits of R@20. Top (`XACI3`): WorldWise++ recovers 12 of the 13 ground-truth predicates and 3 of the 4 of the unobserved door, against 7 and 2 for W-DSGDetr++. Bottom (`X37P1`): W-DSGDetr++ recovers 13 of 16 against 12, and 2 of the 3 predicates of the unobserved clothes against 1.

Table 22: **Component-wise ablation of WorldWise (PredCls, DINOv3-L).** The study is not scored on the 1,511-video test split of Table 2, so its rows are compared only with one another. Rows prefixed with − remove one component from the full model; + VLM supervision adds VLM pseudo-label training (λ_{vlm}=0.2) to the full model; τ=0.75 raises the logit-adjustment temperature from its default τ=0.5.

	With Constraint (WC)				No Constraint (NC)			
Method	**R@K**		**mR@K**		**R@K**		**mR@K**	
	@10	**@20**	**@10**	**@20**	**@10**	**@20**	**@10**	**@20**
WorldWise (full)	63.37	69.17	40.11	49.47	74.50	92.62	60.67	81.75
+ VLM supervision (λ_{vlm}=0.2)	60.67	66.23	34.57	41.54	71.69	90.64	52.81	76.02
τ=0.75 (mR-max)	62.04	67.76	**42.33**	**52.84**	72.47	91.33	**65.44**	**83.43**
− logit adjustment	**64.71**	**70.52**	34.46	41.65	**76.28**	**93.97**	49.58	74.61
− artificial masking	64.16	70.02	39.77	48.95	75.16	93.15	59.97	81.22
− EMA recon. target	63.47	69.23	39.49	48.98	74.42	92.79	60.68	81.61
− pair geometry	63.56	69.30	37.61	46.52	74.48	92.94	58.74	80.84
− ego-motion enc.	63.98	69.77	38.52	47.45	74.90	93.07	59.69	81.39
− object motion enc.	63.71	69.50	39.69	48.95	74.91	93.01	59.61	79.96
− temporal edge attn.	62.56	68.29	40.00	49.41	73.97	92.64	60.24	81.83

G.6 PREDICATE COMPOSITION BY VISIBILITY BUCKET

Table 23 counts the expanded predicate instances of a 1,525-video version of the test annotation by visibility bucket: 444,550 involve an observed object (OO) and 74,407 an unobserved object (UO). Each predicate label attached to a human–object pair at a timestamp counts once. Because the human is continuously observed, every instance belongs to the OO or the UO bucket. These counts are neither the object-coverage counts nor the bucket denominators of the 1,511-video test split, whose support per bucket and per predicate Tables 20 and 21 report.

Table 23: **Expanded test predicate instances by visibility bucket** (1,525-video version of the test annotation).

Visibility Category	# GT Triplets	Share (%)
Observed objects (OO)	444,550	85.7%
Unobserved objects (UO)	74,407	14.3%
Total	518,957	100.0%

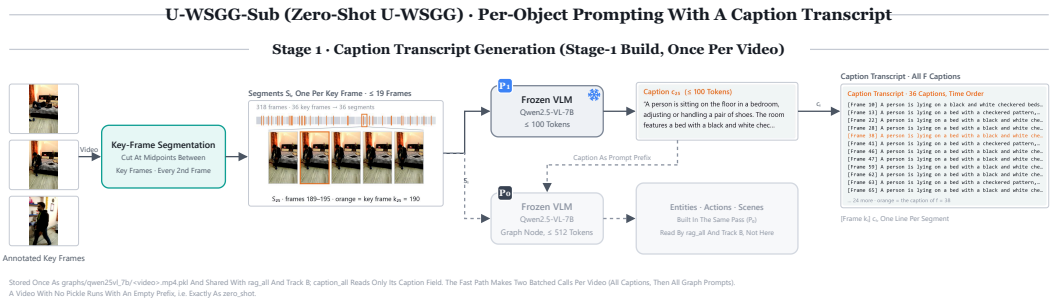


Figure 39: **U-WSGG-Sub, Stage 1: Caption Transcript Generation (Stage-1 Build, Once Per Video).** The text context of U-WSGG-SUB is built offline, once per video. The annotated frames serve as key frames, and the raw frame range is cut at the midpoints between consecutive key frames, so segment i is centred on annotated frame f_i; every second raw frame of a segment is kept and resized to a 128,000-pixel budget, giving a clip tensor S_i of at most 19 frames. The frozen VLM captions each clip ($\leq$ 100 new tokens), turning pixels into a one-sentence summary of the main action; the captions, each tagged with its annotated frame, form the caption transcript of the video, which is the only representation this method keeps (the structured graph built from the same clips is discarded). The panels show the key-frame segments of test video `00T1E`, the caption generated for each segment and the resulting transcript. All panels are real intermediates of the Qwen2.5-VL-7B run (`--skip-verification`) on `00T1E`.

H ZERO-SHOT WSGG AND U-WSGG: METHODS, EVALUATION PROTOCOLS, AND RESULTS

H.1 ZERO-SHOT U-WSGG METHODS

The zero-shot U-WSGG methods are training-free. For every annotated frame they emit a class-only scene graph: a person, a set of named objects, and person→object attention, contacting and spatial predicates; no method in this family emits a box. The two methods share one per-object relationship prompt and differ in the text context prepended to it: U-WSGG-SUB prepends a caption transcript of the video, and **UWSGG-GRAPHRAG** prepends the node of a video-level event graph retrieved for the object. Both are implemented on one base class that owns the frame loading, the SGDet object discovery, the response parser and the verification pass, so each pipeline below is complete in itself and the two differ only where stated. The full-split rows use Qwen2.5-VL-7B-Instruct for every VLM call through the vLLM engine (tensor parallelism, relationship and node-check prompts submitted in chunks of 64), with standard decoding at temperature 0.2. Each method's evaluation protocol is in Appendix H.3.1.

H.1.1 U-WSGG-SUB: CAPTION-ENRICHED PER-OBJECT PROMPTING

U-WSGG-SUB prepends a natural-language transcript of the video to the per-object relationship prompt and changes nothing else. The caption transcript comes from the same offline Stage-1 build that **UWSGG-GRAPHRAG** reads as a graph, so the two methods look at the same VLM description of the video, once as free text and once as a retrievable structure. The complete pipeline follows.

Offline Stage-1 build: captions and clip graphs. The build runs once per video and model. The key frames are the annotated frames; no model selects them. The raw frame range is cut at the midpoints between consecutive annotated frames, so segment i is centred on annotated frame f_i; the first segment starts at most 30 frames before f_1 and the last ends at most 30 frames after f_K. Every second raw frame of a segment is kept and resized to a total budget of 128,000 pixels, which caps the segment tensor S_i at 19 frames. Per segment the VLM is called twice, caption first. The caption prompt (at most 100 new tokens) reads

```
You are provided with a video clip.  Your task is to
generate a concise and descriptive caption for this clip.
The caption should summarize the main action or event
occurring in the video.  Focus on the visual content and
avoiding hallucinating details not present in the video.
```

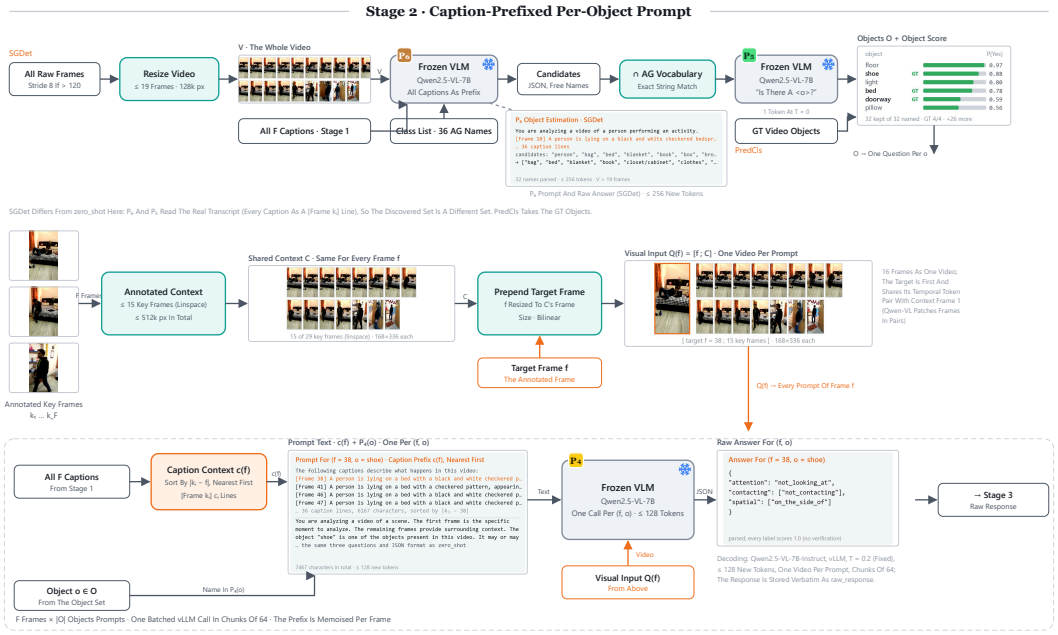
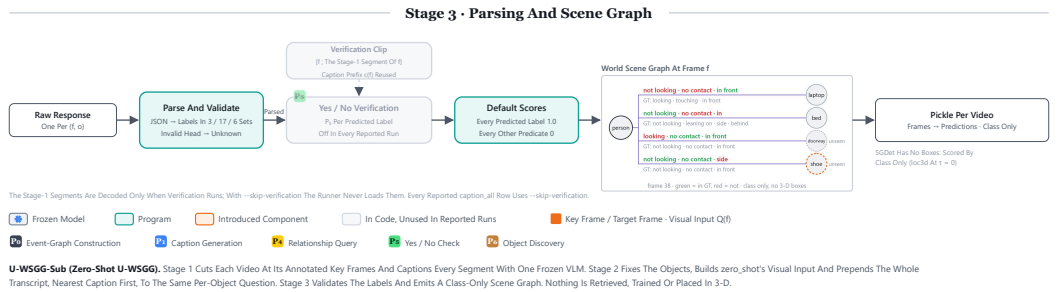


Figure 40: **U-WSGG-Sub, Stage 2: Caption-Prefixed Per-Object Prompt.** For every annotated target frame f_k the transcript is re-ordered by temporal proximity, $|i_s - k|$, so the captions nearest the target come first, and formatted into the context prefix $\textsc{SubCtx}(\mathcal{S}, f_k)$; the prefix is computed once per frame and shared by all objects queried on it. The visual input is the query tensor Q_f, the target frame followed by at most 15 annotated frames chosen by `linspace` (512,000 pixels in total), which gives the model the moment to analyse and the rest of the video as context. The prompt is the prefix followed by the per-object relationship query $\textsc{RelQuery}(o)$, which states that the object may or may not be visible at the target frame and asks for exactly one attention label, one or more contacting labels and one or more spatial labels in JSON. All frame–object prompts of a video are answered in one batched call. In SGDet the object set is first discovered from the whole-video tensor with the transcript as context, and each object's $P(\text{Yes})$ becomes its confidence. The panels show, for one target frame, the captions sorted into the prefix, the query tensor Q_f and the prompt. All panels are real intermediates of the Qwen2.5-VL-7B run (`--skip-verification`) on `00T1E`.



U-WSGG-Sub (Zero-Shot U-WSGG). Stage 1 Cuts Each Video At Its Annotated Key Frames And Captions Every Segment With One Frozen VLM. Stage 2 Fixes The Objects, Builds zero_shot's Visual Input And Prepends The Whole Transcript, Nearest Caption First, To The Same Per-Object Question. Stage 3 Validates The Labels And Emits A Class-Only Scene Graph. Nothing Is Retrieved, Trained Or Placed In 3-D.

Figure 41: **U-WSGG-Sub, Stage 3: Parsing And Scene Graph.** The raw answer is parsed as JSON (with a search for the first balanced $\{\ldots\}$ as fallback) and each head is validated against its label set (3 attention, 17 contacting, 6 spatial labels); an invalid head becomes `unknown`. With verification skipped, every predicted label scores 1.0, and the predictions of all objects of a frame form its class-only scene graph (person, named objects, predicates, no boxes), which the shared adapter maps onto the evaluator's object representations. The panels show the raw answer, its parse, the scene graph of the frame and the ground truth. For `00T1E` the transcript has 36 captions; the caption of the target frame comes first in the context, and the real prompt at (frame 38, shoe) is 7,467 characters long. The answer there is `not_looking_at` / `not_contacting` / `on_the_side_of`, and the caption-driven SGDet discovery names all 4 ground-truth objects. The Stage-1 graph node is built but unused by this method. All panels are real intermediates of the Qwen2.5-VL-7B run (`--skip-verification`) on `00T1E`.

and the graph prompt (at most 512 new tokens), sent with the prefix `Context: The following caption describes this video clip: "<caption>"`, asks for strict JSON of the form `{"entities": [{"entity name", "description"}], "actions": [{"entity name", "action description"}], "scenes": [{"location"}]}`, with action entity names constrained to the entity list and a first-person description when the video is shot from a first-person viewpoint. Each segment is saved as one record with its clip metadata (annotated frame, start and end frame), its caption, and a one-node graph carrying four lists (entities as `"name, description"` strings, actions as `"name, action description"` strings, scene locations, and the caption). The fast path that produced the test-split build issues two batched calls per video, all captions and then all graph prompts, and retries an unparseable graph response up to four times. Every Stage-1 graph of the reported runs comes from Qwen2.5-VL-7B. U-WSGG-SUB reads only the `caption` field of each record and discards the graph.

Inputs and object set. Three visual tensors are built: the whole-video tensor V (all raw frames, stride 8 above 120, subsampled uniformly to at most 19 frames at a 128,000-pixel budget; SGDet discovery only), the per-frame clips S_i (every second raw frame of the segment around annotated frame f_i, at most 19; verification only) and the query tensor $Q_f = f \oplus \mathbf{A}$ (the target frame followed by at most 15 annotated frames chosen by `linspace`, resized to a 512,000-pixel budget, about 168×336 per frame; one video per prompt), so the visual context of every query is $\textsc{VisCtx}(f_k) = Q_{f_k}$. In PredCls the object set is the annotated non-person labels of the video. In SGDet the object-estimation prompt (`You are analyzing a video of a person performing an activity. {caption_context} Here is a list of candidate objects [...] select ALL objects from the candidate list (excluding "person") that the person interacts with [...] Respond ONLY with a JSON list of object name strings.`, at most 256 new tokens, over V with the 36-class foreground candidate list (including person, which the prompt instructs the model to exclude)) is used with the real transcript as its caption context, one line `[Frame f_i] caption_i` per segment. The answer is lower-cased and intersected with the canonical vocabulary by exact string match; the ground-truth list is never consulted. Each surviving object is then scored with the discriminative query `Is there a {object} in this scene? Answer only Yes or No.`, with the same caption context prefixed and V attached, decoding one token at temperature 0 and computing $p_{\text{yes}}(o) = p_{\text{Yes}}/(p_{\text{Yes}} + p_{\text{No}})$ from the top-20 log-probabilities. These scores do not filter the set; they become the object confidence that SGDet scoring multiplies into every triplet score.

Caption context. Generated captions (called subtitles in the implementation) $\mathcal{S} = \{(i_s, t_s)\}$ pair an annotated frame index with a text summary. For each annotated frame f_k, they are sorted by temporal proximity and formatted into a context string,

$$\textsc{SubCtx}(\mathcal{S}, f_k) = \text{format}\big(\text{sort}(\mathcal{S},\ \text{key} = |i_s - k|)\big), \tag{68}$$

which reads

```
The following captions describe what happens in this video:
[Frame 000012] A person is drinking from a cup.
[Frame 000048] ...
```

and is prepended to the per-object relationship prompt:

$$\hat{R}_{\text{sub}}(f_k, o_j) = \text{VLM}\big(\textsc{SubCtx}(\mathcal{S}, f_k) \oplus \textsc{RelQuery}(o_j),\ \textsc{VisCtx}(f_k)\big). \tag{69}$$

$\textsc{RelQuery}(o_j)$ is the per-object prompt: it states that the first frame is the moment to analyse and the remaining frames are context, that the object `"{object_name}"` may or may not be visible in the target frame while a person is, and asks three questions, attention (exactly one of the 3 labels), contacting (one or more of the 17) and spatial (one or more of the 6, the object's position relative to the person), answered as `{"attention": "<label>", "contacting": [...], "spatial": [...]}`. The context string is identical for all objects queried on the same frame; it is computed once per frame and reused across the M per-object prompts. A video without a Stage-1 pickle runs with an empty prefix.

Algorithm 3 U-WSGG-SUB: Caption-Enriched VLM Querying

Require: Video V, frames $\mathcal{F}$, objects $\mathcal{O}_v$, subtitles $\mathcal{S}$
Ensure: Predictions $\{\hat{R}(f_k, o_j)\}_{k,j}$

$\mathbf{T}_V \leftarrow \text{LOADVIDEO}(V)$
$\mathbf{A} \leftarrow \text{BUILDANNOTATEDCTX}(\mathcal{F}, V)$
$\mathcal{S} \leftarrow \text{LOADSUBTITLES}(V)$ ▷ From the offline Stage-1 build
for $f_k \in \mathcal{F}$ **do**
  $\mathbf{v}_k \leftarrow f_k \oplus \mathbf{A}$
  $s_k \leftarrow \text{SUBCTX}(\mathcal{S}, f_k)$ ▷ Proximity-sorted, cached per frame
end for
$\mathcal{Q} \leftarrow \emptyset$
for $f_k \in \mathcal{F},\ o_j \in \mathcal{O}_v$ **do**
  $p \leftarrow s_k \oplus \text{RELQUERY}(o_j)$ ▷ Subtitle + query
  $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{(p,\ \mathbf{v}_k)\}$
end for
$\{\hat{R}\} \leftarrow \text{VLM}_{\text{batch}}(\mathcal{Q})$ ▷ 1 batched call
$\{p_{\text{yes}}\} \leftarrow \text{BULKVERIFY}(\{\hat{R}\})$ ▷ Optional; skipped in the reported runs

Generation, parsing and verification. All $K \times M$ prompts are processed in one batched vLLM call (chunks of 64, one video per prompt, at most 128 new tokens), parsed as JSON (with a greedy search for the first balanced $\{\dots\}$ as fallback) and validated against the 3 / 17 / 6 label sets, an invalid head becoming `unknown`. Verification, when enabled, re-asks each predicted label as a Yes/No question with the same caption context reused as the verification prefix and the visual input $[f_k; S_k]$; the reported runs skip it, so every predicted label scores 1.0. The output is one pickle per video holding, per frame, the object list and one prediction per object with the raw response and the three scored label sets; a shared adapter converts it to the evaluator's record format.

Limitations. Generated captions provide unstructured scene context that does not adapt to the specific object being queried: a caption describing "the person picks up a cup and walks to the table" is equally prepended whether the query targets the cup, the table, or a chair that is out of frame. The captions are also produced by the model being evaluated, so the caption rows conflate captioning ability with relational ability. This motivates the structured, query-specific retrieval of **UWSGG-GRAPHRAG**.

H.1.2 UWSGG-GRAPHRAG: GRAPH-STRUCTURED RETRIEVAL-AUGMENTED PROMPTING

UWSGG-GRAPHRAG is the method the paper means by Graph RAG (Figure 4 in the main paper). Its hypothesis is object permanence: a relationship question about an object the camera cannot see at the target frame should be answerable from a structured record of what happened elsewhere in the video. U-WSGG-SUB gives that record as prose and hands the model all of it; **UWSGG-GRAPHRAG** gives it as a graph and hands the model only the part that matches the query. Two stages share one frozen VLM; retrieval is performed by a frozen text encoder and never by the VLM. The complete pipeline follows.

Stage 1: offline event-graph construction. The offline build runs once per video and model. The key frames are the annotated frames $f_1 < \cdots < f_K$. Segment i is bounded by the midpoints to f_{i-1} and f_{i+1}, with the first and last segments clipped to 30 frames around their key frame; every second raw frame is kept and resized to a 128,000-pixel budget, giving a tensor S_i of at most 19 frames. Two VLM calls are made per segment: the caption prompt ($\leq$ 100 new tokens; "generate a concise and descriptive caption [...] summarize the main action [...] avoiding hallucinating details") and then the graph prompt ($\leq$ 512 new tokens), prefixed with `Context: The following caption describes this video clip: "<caption>"`, which returns strict JSON `{"entities": [{"entity name", "description"}], "actions": [{"entity name", "action description"}], "scenes": [{"location"}]}` with action entity names constrained to the entity list. Node i is $\{\text{entities}_i, \text{actions}_i, \text{scenes}_i, \text{captions}=[\text{caption}_i], \text{key frame } f_i\}$. Within a node, every item is embedded with BGE and linked to an existing entity key when their cosine similarity exceeds 0.7; because each segment is a single node these links are self-loops, and Stage 2 never reads the stored edges. The links between events that a diagram draws exist as Stage 2's entity index: two nodes are

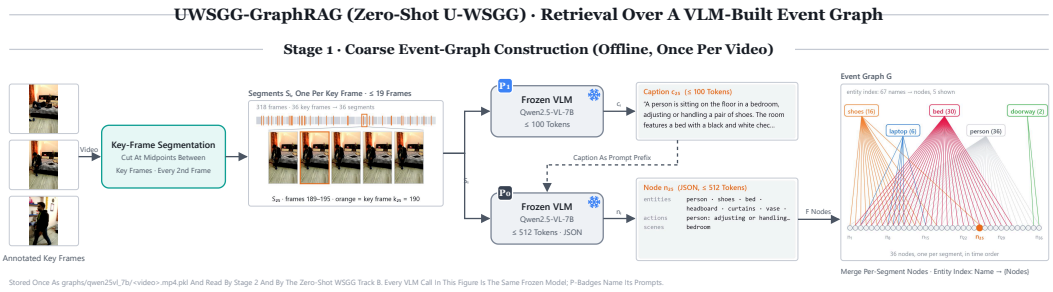


Figure 42: **UWSGG-GraphRAG, Stage 1: Coarse Event-Graph Construction (Offline, Once Per Video).** The video is cut into key-frame segments exactly as for U-WSGG-SUB (midpoints between annotated frames, every second frame, at most 19 frames at 128,000 pixels). For each segment the frozen VLM first writes a caption and then, with that caption as context, a strict-JSON graph of the clip: its entities with descriptions, the actions of those entities, and the scene locations. Each segment becomes one event node {entities, actions, scenes, caption, key frame}, so the video is turned from pixels into a set of structured, time-stamped records that can later be searched by object; a video without nodes is skipped. The panels show the key-frame segments of test video `00T1E`, the caption card of one segment and the entity/action/scene node built from it. All panels are real intermediates of the Qwen2.5-VL-7B run (`--skip-verification`) on `00T1E`.

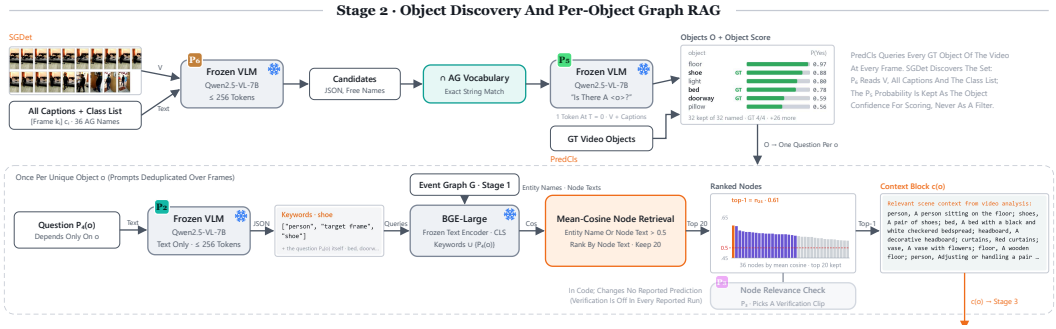
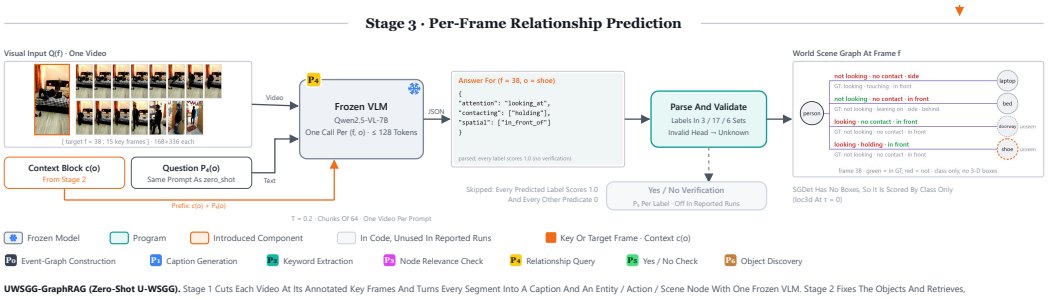


Figure 43: **UWSGG-GraphRAG, Stage 2: Object Discovery And Per-Object Graph RAG.** The event nodes are merged into one video-level graph, and an entity index maps every entity, action and scene key to the nodes that mention it, which links events that share an entity; the frozen BGE text encoder embeds every entity key and every node text once per video. In SGDet the object set is first discovered from the whole-video tensor V with the captions as context. Retrieval then runs once per object, not per frame: (1) a text-only VLM call extracts keywords from the relationship question; (2) the keywords and the question are embedded, nodes are selected through the entity index (mean cosine above 0.5 to an entity key) and by their own text (mean cosine above 0.5), ranked by cosine similarity and the top 20 kept; (3) the top-1 node's entities, actions and scenes form the object's context block. The block turns the unstructured video record into the few facts that concern the queried object, including those observed at other times. The panels show the SGDet discovery over V, and for one object (the shoe, chosen because it has parsed keywords and the most unobserved frames) the keyword-extraction card, the event graph with the entity index, the BGE cosine ranking of the nodes with the retained top-20, and the context block of the top-1 node. All panels are real intermediates of the Qwen2.5-VL-7B run (`--skip-verification`) on `00T1E`.



UWSGG-GraphRAG (Zero-Shot U-WSGG). Stage 1 Cuts Each Video At Its Annotated Key Frames And Turns Every Segment Into A Caption And An Entity / Action / Scene Node With One Frozen VLM. Stage 2 Fixes The Objects And Retrieves, Per Object, The Node Closest To Its Question Under A Frozen Text Encoder. Stage 3 Asks The Same VLM For Attention, Contacting And Spatial Predicates At Every (Frame, Object). Nothing Is Trained Or Placed In 3-D.

Figure 44: **UWSGG-GraphRAG, Stage 3: Per-Frame Relationship Prediction.** Every (frame, object) pair is answered by one VLM call: the object's context block is prepended to the per-object relationship query, and the visual input is the query tensor Q_f of the target frame (the target frame followed by at most 15 annotated frames). The block is the same at every frame of the object, so temporal specificity comes only from the target frame in Q_f. The answer is parsed as JSON and validated against the 3 / 17 / 6 label sets, and the per-object predictions of a frame form its class-only scene graph. The panels show the query tensor of frame 38, in which the shoe is unobserved, the prompt with the context block, the raw answer (`looking_at` / `holding` / `in_front_of` at (38, shoe)) and the resulting scene graph against the ground truth. The segments are sampled at every second frame ($\leq$ 19 frames); the P1/P0/P2/P6 token budgets are printed on the figure; retrieval uses BGE CLS embeddings, a 0.5 threshold and the top-20 nodes. The node relevance check and the verification call are drawn ghosted because they change no reported prediction. All panels are real intermediates of the Qwen2.5-VL-7B run (`--skip-verification`) on `00T1E`.

linked when they mention the same entity name. The build is a hard gate: a video with a missing or zero-node graph is skipped by **UWSGG-GRAPHRAG**.

Stage 2 inputs: event graph, entity index and embeddings. The K per-segment graphs are merged into one video-level directed graph; node ids are offset by the running node count to avoid collisions and each node is tagged with its source annotated frame, which later maps a retrieved node back to its clip. An entity index is built as an inverted index: for every entity, action and scene string of a node, the lower-cased text before the first comma becomes a key that maps to that node. The caption list $[(f_i, \text{caption}_i)]$ and the segment intervals are returned as well. Graph embeddings are computed once per video with the frozen BGE-large-en-v1.5 encoder: every entity key is embedded, and every node is embedded from its text formed as `entities; actions; scenes; scenes; captions` (the scenes appear twice, a weighting property of the implementation). Embeddings are the L2-normalised CLS vectors, so cosine similarity is a dot product.

Visual tensors and object set. Three visual tensors are built: the whole-video tensor V (raw frames, stride 8 above 120, subsampled to at most 19 at 84×140 for a 480×270 source), the segment tensors S_i (the Stage-1 rule), and the query tensor Q_f (target frame followed by at most 15 annotated frames chosen by `linspace`, 512,000 pixels in total, about 168×336 per frame). In PredCls the object set is every non-person label annotated in the video, queried at every annotated frame. In SGDet the object-estimation prompt receives V, every Stage-1 caption as `[Frame f_i] caption_i` lines and the 36-class foreground candidate list (including person, which the prompt instructs the model to exclude), returns a JSON list ($\leq$ 256 new tokens), which is lower-cased and intersected with the vocabulary by exact match; each surviving object is then scored by `Is there a <object> in this scene? Answer only Yes or No.` with the captions prefixed and V attached, decoding one token at temperature 0 and computing $P(\text{Yes}) = p_{\text{yes}}/(p_{\text{yes}} + p_{\text{no}})$ from the top-20 log-probabilities. The scores do not filter the set; they multiply into every SGDet triplet score. Object discovery involves no embedding matching.

Deduplication. The relationship prompt text depends only on the object name, so the $K \times M$ queries reduce to M unique prompts. Keyword extraction, retrieval and the node check run once per unique object; only the final answer runs per (frame, object). Retrieval is therefore per object, not per frame: object o receives the same context block at every frame of the video, and temporal specificity enters only through the target frame in Q_f.

Step 1: keyword extraction. One batched, text-only VLM call ($\leq$ 256 new tokens, no image) sends the reasoning prompt inherited from the Vgent long-video pipeline with the whole relationship prompt as the question and an empty candidate list. It asks the model to "identify the information, like entities, scene, action from the question that is important to retrieve the segments" and to answer in JSON with the fields `keywords`, `multiple`, `time`, `tool`, `candidates_necessary` and `global`; only `keywords` affects retrieval. The response is parsed with `json.loads`; if it does not parse, the keyword list is empty and the query set below reduces to the relationship prompt alone.

Step 2: embedding retrieval (no VLM call). The query set is $\mathcal{Q}_o$ = keywords$_o$ $\cup$ $\{\textsc{RelQuery}(o)\}$, embedded with BGE. Nodes are selected by two routes and ranked once. *Entity route:* each entity key receives its mean cosine similarity to the queries in $\mathcal{Q}_o$; every node indexed under a key scoring above 0.5 is selected. *Content route:* any other node whose node text has a mean cosine above 0.5 is added. The selected nodes are ranked by the mean cosine between $\mathcal{Q}_o$ and the node text and the top $N_r = 20$ are kept. If nothing clears the threshold the object receives no context and no node checks. The Vgent special cases keyed on quoted captions, "beginning", "at the end of the video" and "order" never fire on the WSGG prompt.

Step 3: node relevance check. For each unique object and each retrieved node n, one VLM call over the node's own segment tensor S_n ($\leq$ 256 new tokens) asks two templated Yes/No sub-questions with the node's caption: `Is the '<o>' visible or interacted with in this video segment?` and `Is there a person visible in this video segment?`; all node-check prompts are processed in one batched call. Nodes are ranked by the number of answers that are not "no". This ranking has exactly one use: it selects the verification clip for a frame that has no Stage-1 segment of its own. The context block below comes from the embedding ranking of Step 2, not from this one; since every annotated frame has its own segment and every reported run skips verification, this step does not change a reported prediction.

Step 4: frame-specific answer. The context block is built from the top-1 node of the embedding ranking, joining its three lists with "; " and omitting its caption:

```
Relevant scene context from video analysis:
<entities>; <actions>; <scenes>
<the per-object relationship prompt:  the first frame is the
moment to analyse, the object "{object_name}" may or may not
be visible while a person is; attention exactly one of 3,
contacting one or more of 17, spatial one or more of 6; JSON
answer>
```

Each (frame, object) pair is answered with this text and the visual input Q_f in one batched VLM call over all pairs (chunks of 64, $\leq$ 128 new tokens; 4,096 for the thinking cells). The answer is parsed as JSON and validated against the 3 / 17 / 6 label sets; an invalid head becomes `unknown`. Note the asymmetry with U-WSGG-SUB: the **UWSGG-GRAPHRAG** generation prompt carries graph context but no captions, whereas U-WSGG-SUB's carries captions but no graph. Captions re-enter the **UWSGG-GRAPHRAG** path only through the retrieval embeddings, the node check, the SGDet discovery prompt and the verification prefix.

Verification and output. Verification, when enabled, asks one Yes/No question per predicted label (`At the target frame (the first frame of this clip), is the person <label> the <object>? Answer only Yes or No.`), prefixed with all captions and the object's context block, with visual input $[f; S_f]$, and would replace the score by $P(\text{Yes})$; every reported run skips it, so each predicted label scores 1.0. The per-video pickle has the same schema as that of U-WSGG-SUB; the adapter maps each prediction onto the frame's ground-truth object representations, adds discovered SGDet objects without a ground-truth match as extra object representations, and sets `has_pred_3d` to false everywhere.

Backbones and decoding. The full-split rows use Qwen2.5-VL-7B-Instruct for every call in both stages, standard decoding at temperature 0.2, at most 128 new tokens for the answer. The matched 150-video block runs Stage 2 on Qwen3-VL-8B Bai et al. (2025) in standard mode and in thinking mode (4,096 new tokens, temperature 0.6, top-p 0.95), reading the same Qwen2.5-VL Stage-1

graphs so that the retrieval corpus is fixed across cells. In thinking runs the keyword response contains a reasoning trace that the parser does not strip, so the keyword list is empty and the retrieval query reduces to the relationship prompt alone; the 4,096-token budget also applies to the node check and the SGDet discovery prompt, while Stage 1 is not re-run.

Calls per video. With K annotated target frames (32.3 on average), K_s cached Stage-1 segments and M objects: $2K_s$ offline prompts before retries (shared with U-WSGG-SUB and Track B), $1 + M$ discovery/confidence prompts in SGDet, M keyword prompts, no VLM prompts for embedding retrieval, at most $20M$ node-check prompts, and KM relationship-answer prompts. Each node-check prompt contains both Yes/No questions. Label verification contributes zero prompts in reported runs and would add one per predicted label if enabled. These are prompt counts, not batch counts; Appendix H.3.3 separates the stages and their batching.

Reading the `00T1E` panels. Figures 39–44 are drawn from sandboxed runs of the headline commands on the video `00T1E` (29 annotated frames; bed, doorway, laptop and shoe; 51 visible and 65 unobserved object representations) with Qwen2.5-VL-7B and `--skip-verification`. The object shown is the shoe, the object with parsed keywords and the most unobserved frames, and the frame is 38, in which the shoe is unobserved. The two methods receive the same query tensor Q_{38} and the same relationship prompt; they differ only in the text prefix, and their answers at (38, shoe) differ accordingly: `not_looking_at` / `not_contacting` / `on_the_side_of` (36-caption transcript, 7,467-character prompt) and `looking_at` / `holding` / `in_front_of` (top-1 retrieved node). In SGDet the caption-driven discovery prompt of U-WSGG-SUB names all 4 ground-truth objects. In the Graph-RAG retrieval panels (Figure 43) the keyword JSON did not parse for the bed and the doorway (the model wrote `'none'` in single quotes), so their retrieval ran on the question alone, and the entity key `person` scores above 0.5 for every question and appears in all 36 nodes, so every node is a candidate and the retained 20 are simply the closest by node text.

H.2 ZERO-SHOT WSGG METHODS

The zero-shot WSGG methods are training-free architectures that give a prompted VLM a canonical world frame. Their scene graphs carry an oriented 3D box per object, so they can be matched geometrically and checked by a program. Both consist of a perception layer of frozen tools and annotation-independent caches, a localized prompt in which the target frame, a top-down map and a numerical box table carry the same object ids, and, in Track B, a fixed tool loop with a geometric critic and one repair call. The two tracks share one implementation of the payload builder, parser and model loader, so they are one architecture with an optional loop; the perception layer and the prompt are described in full under each method. The evaluation of both methods and the critic analysis are in Appendix H.4.1, and their SGDet results by visibility bucket in Appendix H.5.

The canonical frame. Every geometric quantity of both methods lives in one canonical frame, in stored reconstruction units: points, cameras, boxes, the map, the critic's checks and the 3D evaluator. The frame is z-up with the floor at $z = 0$. π^3 output is scale- and affine-ambiguous and has no reference frame, so everything passes through the corrected transform chain $T_{XY} \circ T_\delta \circ T_{\text{auto}}$ of the annotation pipeline, exposed as $\mathbf{C}_{\text{final}} = \mathbf{A}_{\text{world}\to\text{final}}(\mathbf{C}_{\text{world}} - \mathbf{o})$. A box is a floor-parallel OBB $(\text{center}, \text{size} = (l, w, h), \text{yaw})$, with yaw the rotation of the length axis about $+z$ and its 8 corners listed bottom face first. π^3 frame k is annotation frame `sampled_frames_idx`$[\text{start} + k]$, with the start offset recovered per video by pose matching. The camera's heading is the azimuth of its optical axis in this frame.

Unit convention in the executed prompts. The loader reads `corners_final` and derives box parameters without applying the separately stored physical scale. The numeric geometry and critic thresholds therefore remain in reconstruction units, denoted **u** below. The executed prompts and saved visual panels nevertheless call these values "metric" or "metres" and display `m`; the verbatim quotations and panel labels retain that wording for reproducibility. Those labels do not establish a conversion to physical metres. This unit mismatch is a limitation of the reported localized runs, and interpreting the critic's thresholds as physical distances would require applying the scale consistently and rerunning them.

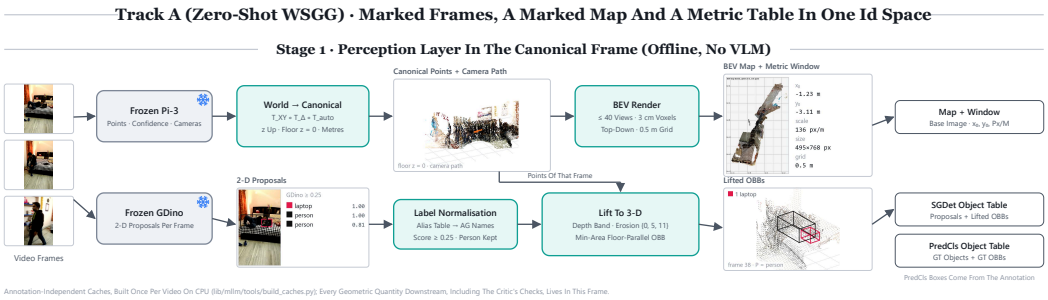


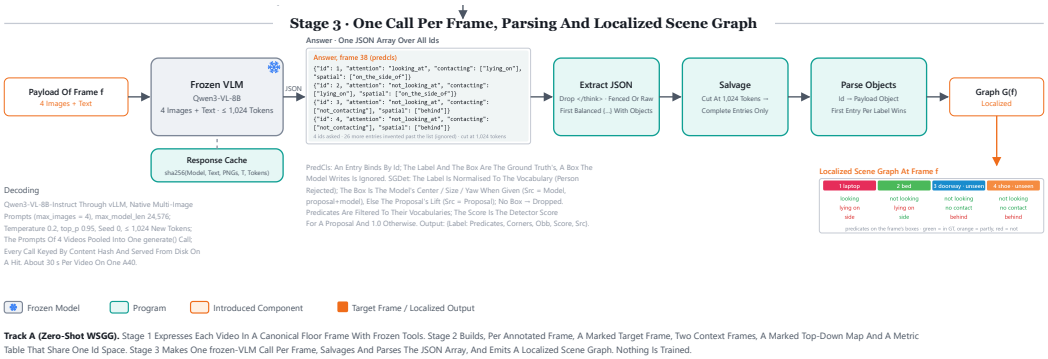
Figure 45: **Track A, Stage 1: Perception Layer In The Canonical Frame (Offline, No VLM).** The perception layer is built offline per video, without a VLM, and gives every object a location in one canonical frame (z up, floor at $z = 0$, reconstruction units). The π^3 reconstruction is moved into this frame; at most 40 of its frames are sampled, confident points are averaged per 0.03 u voxel, and a top-down splat with a 0.5 u grid and the camera path becomes the bird's-eye base map. Grounding DINO proposals of each sampled frame are normalised onto the ActionGenome vocabulary (scores below 0.25 dropped), and each 2D box is lifted to a floor-parallel oriented box from the π^3 points inside it: a depth band around the median depth, several erosion kernels, a percentile trim and a minimum-area rectangle, keeping the smallest candidate; a box with too few points is stated as `3D box unknown`. The panels show, for test video `00T1E`, the π^3 point cloud in the canonical frame, the rendered bird's-eye base map with its grid and camera path (the saved labels say metres), the Grounding DINO proposals of the target frame and their lifted oriented boxes. All panels are real intermediates of the Qwen3-VL-8B run ($\leq$ 1,024 new tokens, temperature 0.2, top-p 0.95, seed 0, four videos pooled, context length 24,576) on `00T1E`.



Figure 46: **Track A, Stage 2: Localized Prompt With Shared Ids (The Payload Builder).** The payload builder gives the VLM one frame of reference in three redundant encodings that share one id per object. (1) The marked target frame outlines every 2D box and tags it `<id> <label>`, draws the person as `P`, and lists objects without a box, the unobserved ones, in a legend as `(not visible here)`; (2–3) two unmarked context frames supply the rest of the video; (4) the marked bird's-eye view stamps each object's footprint with the same id and draws the target camera as a red arrow. The text states the coordinate conventions (with the original metre labels) and the predicate vocabulary, explains the four images, and lists a numerical box table with the person's box, the camera position and heading, and one line per id with label, visibility and oriented box, followed by the task and a one-line JSON schema. In PredCls the ids are the frame's ground-truth objects with their annotated boxes; in SGDet they are the proposals with their lifted boxes and detector scores. The panels show the four images of the PredCls call for frame 38 in prompt order, the marked target frame with the legend of unobserved ids, the two unmarked context frames, the marked bird's-eye view with the camera arrow, and the text of the prompt with the numerical object table. All panels are real intermediates of the Qwen3-VL-8B run ($\leq$ 1,024 new tokens, temperature 0.2, top-p 0.95, seed 0, four videos pooled, context length 24,576) on `00T1E`.

H.2.1 Track A: Marked Frames And Bird's-Eye View

A VLM asked for 3D placement has no frame of reference. Track A supplies one in three redundant encodings that share one id space, marks on the target frame, footprints on a top-down map and rows of a numerical box table, and asks the model to reconcile them in one call per annotated frame. The complete pipeline follows.



Track A (Zero-Shot WSGG). Stage 1 Expresses Each Video In A Canonical Floor Frame With Frozen Tools. Stage 2 Builds, Per Annotated Frame, A Marked Target Frame, Two Context Frames, A Marked Top-Down Map And A Metric Table That Share One Id Space. Stage 3 Makes One frozen-VLM Call Per Frame, Salvages And Parses The JSON Array, And Emits A Localized Scene Graph. Nothing Is Trained.

Figure 47: **Track A, Stage 3: One Call Per Frame, Parsing And Localized Scene Graph.** Each annotated frame is answered by one VLM call over the four images and the text (the frames of four videos pooled into one batch), which must reconcile the marks, the map and the table into a JSON list of objects with their three predicate heads. `extract_json` takes the first balanced object with an `objects` key, `salvage_objects` keeps the complete entries of an answer cut off at the token budget, and `parse_objects` binds each entry: in PredCls by id to a payload object, whose label and box are kept; in SGDet with the label normalised to the vocabulary and the model's box, or the proposal's lifted box when none is given. Predicates are filtered to their vocabularies and one object per label is kept, so the output is a localized scene graph in which every object carries an oriented 3D box. The panels show the raw answer (which in this frame overruns the id list until it is cut at the 1,024-token budget, as in 26 of the 29 frames), the Extract JSON → Salvage → Parse chain with the PredCls binding rule, and the localized scene graph against the ground truth. All panels are real intermediates of the Qwen3-VL-8B run ($\leq$ 1,024 new tokens, temperature 0.2, top-p 0.95, seed 0, four videos pooled, context length 24,576) on `00T1E`.

Perception layer (offline, CPU, no VLM). The caches are built once per video by a single tool and registered as annotation-independent, so an annotation revision only costs new prompts. *Map.* From the π^3 dynamic scene (points, confidences, cameras) transformed into the canonical frame, at most 40 π^3 frames are sampled uniformly; points with confidence ≥ 0.05, every second pixel, and $-2\,\mathrm{u} < z < 3\,\mathrm{u}$ are kept and averaged per 0.03 u voxel, and all camera poses are kept. The map window is the 1st–99th percentile of the point xy, extended by the camera positions, padded by 0.5 u and at least 1 u in extent, at $\min(768/\text{longest side}, 200)$ and at least 60 pixels per reconstruction unit. The base render is a top-down splat in which the highest point wins, brightened with height over 0–2 u, with a 0.5 u grid labelled as metres in the saved render (`x=..`, `y=..`) and the camera path as grey dots. *Proposals.* The precomputed Grounding DINO Liu et al. (2024) boxes of each sampled frame have their free-text labels normalised onto the short ActionGenome vocabulary through an alias table (`glass`/`bottle`→`cup`, `camera`→`phone`, `couch`→`sofa`, …) or exact match; unknown labels and scores below 0.25 are dropped and person is kept. Each box is stored in original and π^3 pixels. *Lift to 3D.* A 2D box becomes an OBB from that frame's π^3 points in the canonical frame: (i) the median camera depth d_0 over the central half of the box anchors a depth band of half-width $\max(0.25\,\mathrm{u}, 0.15\,d_0, 0.6 \times$ the box's size in stored units); (ii) for each erosion kernel in $\{0, 5, 11\}$ pixels the points inside the eroded box and the band with confidence ≥ 0.05 form a candidate, which needs at least 25 points; (iii) points outside the 5th–95th percentile on each axis are dropped; (iv) a floor-parallel minimum-area rectangle is fitted in xy spanning $[z_{\min}, z_{\max}]$; (v) the smallest-volume candidate is kept, and a proposal with no candidate has no box and is stated as `3D box unknown`. Results are cached per (frame, label, rounded box). *Set-of-mark.* The target frame is downscaled to at most 640 pixels on its long side; each 2D box is outlined and tagged `<id> <label>` in a 15-colour palette, the person is drawn in black as `P person`, and objects without a 2D box, i.e. unobserved objects, are listed in a top-left legend as `<id> <label> (not visible here)`. On the map, `mark_bev` stamps the object footprints (OBB bottom faces) with the same numbers, the person's footprint as `P`, and the target-frame camera as a red dot with a 0.5 u arrow along its viewing direction. *Response cache.* Each VLM call is keyed by a SHA-256 over the model key, the prompt text, the PNG hashes of its images, the temperature and the token budget; hits are served from disk.

Object table. In PredCls the ids $1..N$ are the frame's ground-truth objects, the first per short label; each carries its ground-truth OBB from the canonical corners and is stated as `visible in`

`the target frame` or `NOT visible in the target frame` (observed means visible and not sourced from a permanence fill); the person `P` carries its ground-truth OBB and 2D box. In SGDet the ids are the frame's Grounding DINO proposals in descending score with person excluded; each carries its lifted OBB or `3D box unknown` and its detector score; the person is the largest person detection, lifted; the text also states the map extent with the original metre labels and the 35-name object vocabulary.

Images, in prompt order. (1) the marked target frame (≤ 640 px long side); (2–3) two unmarked context frames chosen by `linspace` over the video's other annotated frames, which with two frames means the first and the last, at ≤ 320 px; (4) the marked bird's-eye view (≤ 640 px). The backbone is loaded with $\texttt{max_images} = \texttt{context_frames} + 2 = 4$; Qwen3-VL receives them as separate images placed before the text.

Text. The text runs in this order: the system rules, the image legend, the scene table, the task, and a one-line JSON schema. The system rules fix the conventions in words:

> `You are a careful 3D scene understanding model. Coordinates are metric, in a fixed world frame: z is up, the floor is z = 0, x/y follow the grid drawn on the top-down map (labels "x=..", "y=.." are metres). A box is {"center": [x, y, z], "size": [length, width, height], "yaw_deg": rotation of the length axis about z}. The person is the only subject; every relationship is between the person and one object. Relationship vocabulary (use these exact strings): - attention (exactly one): ... - contacting (one or more): ... - spatial (one or more, position of the OBJECT relative to the PERSON): ...`

The legend explains the four images (`Image 1: the TARGET frame with numbered boxes (P = person). Images 2-3: unmarked frames from the same video, in time order, for context. Image 4: top-down map of the whole room reconstructed from the video (grey dots = camera path, red arrow = camera position and viewing direction at the target frame, numbered polygons = object footprints, P = person footprint).`). The scene table lists `- P person: center=(x, y, z) m, size=(l, w, h) m, yaw=D deg`, the camera position and heading, and one line per id with label, visibility and box. The PredCls task reads: `Task: for EVERY listed object id, decide the person's attention, contacting and spatial relationships with it at the target frame. Use the map and the boxes for objects that are not visible; use the images for what the person is doing. Answer with ONLY this JSON, compact on one line, no markdown, no explanation:` followed by `{"objects": [{"id": 1, "attention": "<label>", "contacting": ["<label>", ...], "spatial": ["<label>", ...]}, ...]}`. The SGDet task reads: `Task: produce the scene graph at the target frame. List every object the person is interacting with or that is relevant to the activity, including objects that are out of view or hidden but whose location you can infer from the video and the map (e.g. the chair the person sat on earlier). For each object give its label, a 3D box in the world frame (copy or correct a proposal's box; estimate a plausible box for objects without a proposal), and the person's attention / contacting / spatial relationships with it. Keep proposal ids when you keep a proposal; use "new" for objects you add.` with the schema extended by `"label"`, `"center"`, `"size"` and `"yaw_deg"`.

Decoding and batching. Track A makes one call per annotated frame with four images. The prompts of four videos are pooled into one vLLM `generate()` call. The headline setting is Qwen3-VL-8B-Instruct, temperature 0.2, top-p 0.95, seed 0, at most 1,024 new tokens and a context length of 24,576. The 150-video thinking cells use the thinking variant with 4,096 new tokens, temperature 0.6 and top-p 0.95.

Parsing. `extract_json` drops any reasoning trace up to the last `</think>`, tries fenced blocks and then the raw text, and takes the first balanced $\{\dots\}$ with an `objects` key; if that fails, `salvage_objects` keeps the complete entries of an array cut off at the token budget (entries never emitted cannot be salvaged). `parse_objects` then binds entries: in PredCls an entry binds by id to a payload object, and the label and box are the ground truth's (any box the model writes is ignored); in SGDet the label is normalised to the short vocabulary with person rejected, the box is the model's centre/size/yaw when given (`src = model` or `proposal+model`), otherwise the proposal's lifted box (`src = proposal`), and an object with neither is dropped because it cannot be scored in 3D. Predicates are filtered to their vocabularies, attention is kept only if valid, one object per label per frame is kept (the first entry wins), and the score is the detector score for proposals and 1.0 otherwise. The output per frame is $\{\text{label} : \{\text{attention}, \text{spatial}, \text{contacting}, \text{corners}, \text{obb}, \text{score}, \text{src}\}\}$ in one pickle per video, read by the shared adapter, which places the person first, the frame's ground-truth objects next and, in SGDet, predicted objects absent from the ground truth last, and sets `has_pred_3d` wherever corners exist.

H.2.2 Track B: Tool Loop With A Geometric Critic

Track B adds to Track A's payload two things: retrieved context from the Stage-1 video graph, and the piece the track exists to test, a geometric critic that inspects the proposed localized scene graph and a repair call that feeds the violations back to the model. A zero-shot U-WSGG (unlocalized) method emits $holding(\text{person}, \text{cup})$ and there is nothing to check it against; a zero-shot WSGG (localized) method emits the same predicate and a box for the cup 2.3 u from the person, which is a contradiction a program can find without a label. The tool order is fixed by the driver, plan $\rightarrow$ perceive $\rightarrow$ retrieve $\rightarrow$ relate $\rightarrow$ verify $\rightarrow$ repair $\rightarrow$ emit; the model never selects a tool. The complete pipeline follows.

Perception layer (offline, CPU, no VLM). The perception layer is built once per video by the cache builder and consists of the map, the proposals, the lift and the set-of-mark renderers. *Map:* the π^3 scene in the canonical frame, at most 40 frames sampled uniformly, points with confidence ≥ 0.05, every second pixel, $-2\,\text{u} < z < 3\,\text{u}$, averaged per 0.03 u voxel; a window at the 1st–99th percentile of the point xy extended by the cameras, padded by 0.5 u, at least 1 u, at $\min(768/\text{longest side}, 200)$ and at least 60 px/u; a top-down splat in which the highest point wins, brightened over 0–2 u, a 0.5 u grid with the original metre labels and the camera path as grey dots. *Proposals:* Grounding DINO boxes with labels normalised onto the short

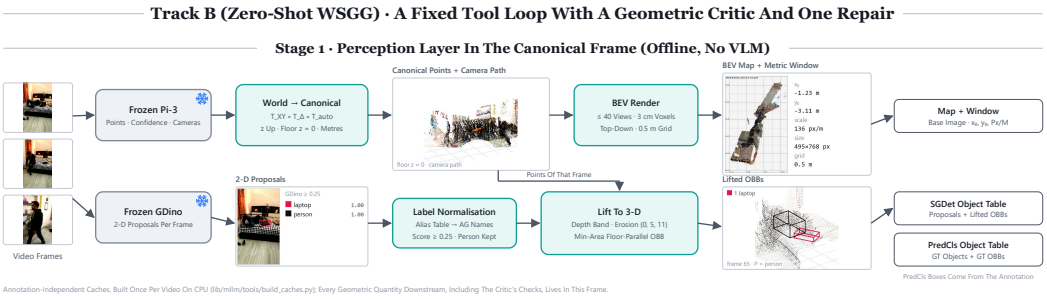


Figure 48: **Track B, Stage 1: Perception Layer In The Canonical Frame (Offline, No VLM).** Track B shares Track A's perception layer. The perception layer is built offline per video, without a VLM, and gives every object a location in one canonical frame (z up, floor at $z = 0$, reconstruction units). The π^3 reconstruction is moved into this frame; at most 40 of its frames are sampled, confident points are averaged per 0.03 u voxel, and a top-down splat with a 0.5 u grid and the camera path becomes the bird's-eye base map. Grounding DINO proposals of each sampled frame are normalised onto the ActionGenome vocabulary (scores below 0.25 dropped), and each 2D box is lifted to a floor-parallel oriented box from the π^3 points inside it: a depth band around the median depth, several erosion kernels, a percentile trim and a minimum-area rectangle, keeping the smallest candidate; a box with too few points is stated as `3D box unknown`. These boxes are also what the geometric critic of Stage 3 checks. The panels show, for test video `00T1E`, the π^3 point cloud in the canonical frame, the rendered bird's-eye base map with its grid and camera path (the saved labels say metres), the Grounding DINO proposals of the target frame and their lifted oriented boxes. All panels are real intermediates of the Qwen3-VL-8B run (same decoding as Track A) on `00T1E`.

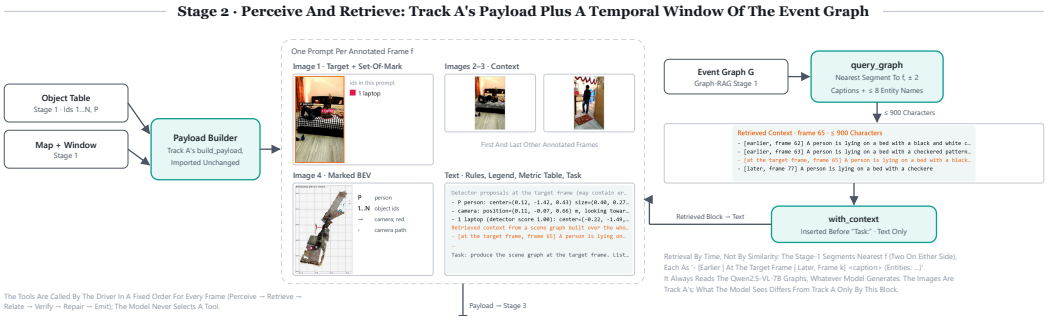
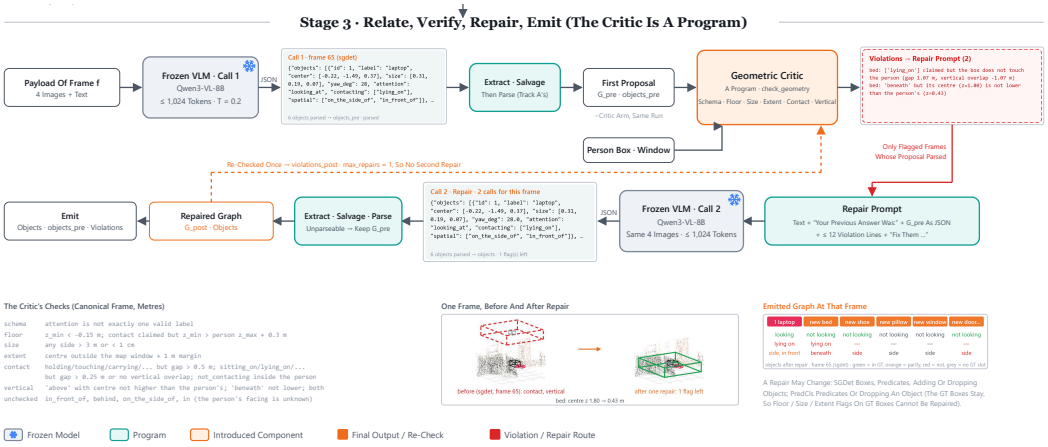


Figure 49: **Track B, Stage 2: Perceive And Retrieve: Track A's Payload Plus A Temporal Window Of The Event Graph.** The proposal round starts from Track A's payload (marked target frame, first and last annotated context frames, marked bird's-eye view; rules, legend, numerical object table, task, schema) and adds text memory of the video. The retrieval reads the Stage-1 event graph of the zero-shot U-WSGG methods by time, without embeddings: the segment nearest the target frame and two on either side are rendered as `[earlier | at the target frame | later, frame k] <caption> (entities: ...)`, cut at 900 characters and inserted before the task, so an object seen earlier and now off-camera remains named in the prompt. One VLM call returns the first proposal, which is parsed as in Track A into the localized graph G_f^{pre}. The panels show the four images, the text with the retrieved-context block and the first proposal. All panels are real intermediates of the Qwen3-VL-8B run (same decoding as Track A) on `00T1E`.



Track B (Zero-Shot WSGG). Stage 1 Expresses Each Video In A Canonical Floor Frame With Frozen Tools. Stage 2 Builds Track A's Four-Image Payload And Inserts The Stage-1 Captions Of The Segments Around The Target Frame. Stage 3 Runs A Fixed Loop: One Proposal Call, A Geometric Critic That Is A Program, One Repair Call For Flagged Frames, One Re-Check. Both Ablation Arms Come Out Of The Same Run.

Figure 50: **Track B, Stage 3: Relate, Verify, Repair, Emit (The Critic Is A Program).** The critic takes the parsed proposal G_f^{pre}, the person box and the map window, sees no image, and checks in metres whether the boxes agree with the claimed predicates: boxes below the floor or implausibly large or small, boxes outside the map, contact predicates claimed with the object more than 0.5 u (support predicates 0.25 u) from the person, and `above`/`beneath` claims that contradict the heights; each violation becomes one sentence. If the critic fires, one repair call re-sends the proposal-round prompt with the previous answer and at most 12 violation lines and asks for a corrected JSON; the repair is parsed, checked again and emitted as G_f^{post} (the proposal is kept when it does not parse). The loop thus adds a label-free consistency check between geometry and relations. The panels show the violations the critic raised on the proposal, the repair prompt, and the SGDet before/after example of frame 65, in which a bed box floating at $z = 1.80$ u is lowered to 0.43 u, clearing the contact violation while the `beneath` check remains flagged. On `00T1E` the SGDet critic flagged 20 of 29 frames (37 violations: 19 size, 9 contact, 9 vertical); the repair lowered the count in 8 frames, cleared it in 1 and raised it in 2. All panels are real intermediates of the Qwen3-VL-8B run (same decoding as Track A) on `00T1E`.

vocabulary by alias table or exact match, scores below 0.25 dropped, person kept. *Lift to 3D:* depth anchoring at the median camera depth d_0 over the box's central half with band half-width $\max(0.25\,\text{u}, 0.15\,d_0, 0.6 \times \text{physical size})$; candidates from erosion kernels $\{0, 5, 11\}$ px with confidence ≥ 0.05 and at least 25 points; 5th–95th percentile trim per axis; floor-parallel minimum-area rectangle in xy spanning $[z_{\min}, z_{\max}]$; smallest-volume candidate kept, otherwise `3D box unknown`. *Set-of-mark:* target frame at ≤ 640 px with each 2D box outlined and tagged `<id> <label>`, the person in black as `P person`, unobserved objects in a top-left legend as `<id> <label> (not visible here)`; the map stamped with the numbered footprints, `P`, and the target camera as a red dot with a 0.5 u arrow. *Response cache:* SHA-256 over model key, prompt text, image hashes, temperature and token budget.

Object table, images and base text. In PredCls the ids are the frame's ground-truth objects (first per short label) with their ground-truth OBBs and a visible / `NOT visible` statement, and the person `P` with its ground-truth OBB and 2D box; in SGDet the ids are the Grounding DINO proposals by descending score with lifted OBBs or `3D box unknown` and detector scores, the person is the largest person detection lifted, and the text adds the map extent and the 35-name vocabulary. The four images are the marked target frame (≤ 640 px), the first and last other annotated frames unmarked (≤ 320 px) and the marked bird's-eye view (≤ 640 px). The text is the system rules (`You are a careful 3D scene understanding model. Coordinates are metric, in a fixed world frame: z is up, the floor is z = 0, x/y follow the grid drawn on the top-down map [...]. A box is {"center": [x, y, z], "size": [length, width, height], "yaw_deg": rotation of the length axis about z}. The person is the only subject [...]` followed by the 3 / 17 / 6 vocabulary with its cardinalities), the image legend (which image is the marked target, which are the unmarked context frames, and what the map's grey dots, red arrow, numbered polygons and `P` footprint mean), the scene table, the task (PredCls: decide the three heads for every listed id, using the map and the boxes for objects that are not visible; SGDet: list every relevant object including those out of view whose location can be inferred, copy or correct a proposal's box or estimate one, keep proposal ids and use `"new"` for added objects) and the one-line JSON schema.

Retrieval: `query_graph`. The retrieval reads the Stage-1 pickles of the zero-shot U-WSGG Graph RAG (Appendix H.1.1), always the Qwen2.5-VL-7B graphs whatever the generation model, so the corpus is fixed. For target frame f the segments are sorted by key frame; the one nearest f and two on either side are taken; each is rendered as `- [earlier | at the target frame | later, frame k] <caption> (entities: ...)`, listing up to eight of the first six entity names of that segment's node in alphabetical order; the block is cut at 900 characters under the header `Retrieved context from a scene graph built over the whole video (may be noisy):` and inserted immediately before the `Task:` line. No embeddings are involved: this is a temporal window, the opposite choice from **UWSGG-GRAPHRAG**, whose retrieval is per object and ignores the frame. The temporal tagging is what carries object permanence in text: "the chair the person sat on earlier" is retrievable even when it is off-camera.

Proposal round. VLM call 1 sends the four images and the text with the retrieved block, for the frames of four videos pooled into one `generate()` call, with Qwen3-VL-8B-Instruct at temperature 0.2, top-p 0.95, seed 0, at most 1,024 new tokens and context length 24,576. The response is parsed by `extract_json` (any reasoning trace up to the last `</think>` dropped; fenced blocks, then raw text; the first balanced $\{\ldots\}$ with an `objects` key), `salvage_objects` (the complete entries of an array cut off at the token budget) and `parse_objects` (PredCls: entries bind by id to the payload objects and keep the ground-truth label and box; SGDet: the label is normalised to the short vocabulary with person rejected, the box is the model's centre/size/yaw when given and otherwise the proposal's lifted box, objects with neither are dropped; predicates filtered to their vocabularies, one object per label per frame, score = detector score for proposals and 1.0 otherwise) into the proposal G_f^{pre}, which is stored as `objects_pre` together with its raw response.

The critic: `check_geometry`. The critic is a program. Its inputs are the parsed graph (boxes and predicates), the person box (the ground truth in PredCls, the lifted detection in SGDet) and the

map window; it sees no image. All checks are in the canonical frame, in stored reconstruction units, and the gap between two boxes is the Euclidean gap between their axis-aligned envelopes (0 when they overlap). It raises a violation of one of the following kinds: *schema*, when the label is not in the vocabulary, attention is not exactly one valid label, or a predicate string is invalid (in practice only the attention rule fires, because parsing already filtered the rest); *floor*, when the box bottom is below the floor ($z_{\min} < -0.15$ u) or a touching-type contact is claimed while the box floats above the person ($z_{\min} >$ person $z_{\max} + 0.3$ u); *size*, when any side exceeds 3 u or is below 0.01 u; *extent*, when the box centre lies outside the map window plus a 1 u margin; *contact*, when `holding`, `touching`, `carrying`, `eating`, `drinking_from`, `wiping`, `writing_on`, `twisting`, `wearing` or `have_it_on_the_back` is claimed with a gap above 0.5 u, when `sitting_on`, `lying_on`, `standing_on`, `leaning_on` or `covered_by` is claimed with a gap above 0.25 u or without vertical overlap, or when `not_contacting` is claimed while the object box lies inside the person box (± 0.05 u); *vertical*, when `above` is claimed with the object centre not higher than the person centre, `beneath` with it not lower, or both are claimed. Without a person box only the schema, floor, size and extent checks run. `in_front_of`, `behind`, `on_the_side_of` and `in` are not checked because the person's facing direction is unknown; the critic therefore covers the height axis of spatial reasoning and not the bearing axis. Each violation is rendered as a sentence such as `cup: ['holding'] claimed but the box is 2.31 m away from the person`.

Repair round. Only frames whose critic fired and whose proposal parsed are re-sent, at most once (`--max_repairs 1`). The repair prompt is the full proposal-round text, followed by `Your previous answer was:` and the proposal re-serialised as JSON, `A geometric checker found these problems with it:` and at most 12 violation lines, and the instruction `Fix them: keep every correct object and relationship, correct the box or the predicate that is inconsistent with the 3D layout (boxes must rest on or above the floor, contact predicates need the object next to the person, 'above'/'beneath' must match the heights). Answer with ONLY the corrected JSON in the same compact format.` The images are the same four. The repaired answer is parsed with the same parser; if it does not parse, the proposal is kept. The repaired graph is checked again, giving `violations_post`. What a repair can change follows from the parser: in SGDet boxes (centre, size, yaw), predicates and the object set (add, drop, relabel); in PredCls predicates and dropping objects only, because ids bind to ground-truth objects whose boxes are kept, so floor, size and extent violations raised on ground-truth boxes persist by design. The repair instruction permits discharging a violation by weakening the predicate rather than moving the box.

Emit and ablation arms. Each frame emits `objects` (G_f^{post}, or G_f^{pre} when no repair ran), `objects_pre`, `violations_pre`, `violations_post`, the retrieved block and the number of calls, in one pickle per video ($\{\text{label} : \{\text{attention}, \text{spatial}, \text{contacting}, \text{corners}, \text{obb}, \text{score}, \text{src}\}\}$ per frame); the shared adapter places the person first, the frame's ground-truth objects next and, in SGDet, predicted objects absent from the ground truth last, and sets `has_pred_3d` wherever corners exist. Both arms of the critic ablation come out of one run: the reported no-critic result scores `objects_pre` from the same run through the usual adapter, so the two arms answer the same first sampled proposal and the comparison measures the critic-plus-repair stage, including its extra call. A `--no_retrieval` arm exists in the code and has not been scored. The calls per video are K proposal calls plus one per repaired frame.

Reading the `00T1E` panels. Figures 45 and 48 are drawn from sandboxed runs of the headline commands on `00T1E` with Qwen3-VL-8B ($\leq$ 1,024 new tokens, temperature 0.2, top-p 0.95, seed 0). Stage 2 of both figures (Figures 46 and 49) shows the PredCls call of frame 38, the frame in which the shoe is unobserved: the payload builder's four images (marked target frame, first and last annotated context frames, marked bird's-eye view) and its text (rules, legend, numerical object table, task, schema); the response cache serves repeated calls. Track A's answer for this frame lists the four real ids and then invents entries up to id 31 until it is cut at the token budget, which happens in 26 of the 29 frames; the parser keeps only the real ids. Track B's Stage 3 example (Figure 50) is chosen by a fixed rule (an SGDet frame whose repair moved a flagged box and lowered the violation count, nearest to frame 38): frame 65, where the bed box at $z = 1.80$ u is lowered to 0.43 u. Over the video the SGDet critic flagged 20 of 29 frames with 37 violations (19 size, 9 contact, 9 vertical);

the single repair round lowered the count in 8 frames, cleared it in 1 and raised it in 2. In PredCls it flagged 3 frames, all contact, and each repair cleared its flag.

H.3 Zero-Shot U-WSGG Evaluation Protocol, Method Comparison and Efficiency

This subsection describes the P/R/F1 evaluator used for the main-paper Table 3. The WorldBBox R@K protocol is defined in Section 5; its zero-shot evaluation scores every annotated object and is distinct from the object-set evaluation below. Caption-enriched and Graph-RAG prompting are compared both in the WorldBBox zero-shot U-WSGG PredCls analysis and in the five-backbone P/R/F1 study. The prompting pipelines themselves are described in Appendix H.1.

H.3.1 Task Definition and Evaluation Protocol

Task Definition. Given a video V with K annotated frames $\{f_1, \ldots, f_K\}$ and a video-level object set $\mathcal{O}_v$, the goal is to predict, for *every* annotated frame f_k and *every* object $o \in \mathcal{O}_v$, the three relationship types between the person and object o:

$$\hat{R}(f_k, o) = \left(\hat{a}_{k,o},\ \hat{C}_{k,o},\ \hat{S}_{k,o}\right) \tag{70}$$

where $\hat{a}$ is the predicted attention label (single-label), $\hat{C}$ is the predicted set of contacting labels (multi-label), and $\hat{S}$ is the predicted set of spatial labels (multi-label). Unlike full WSGG, no 3D localization is required; the object set is either given or must be discovered, as follows.

Object Set Resolution: PredCls vs. SGDet. The object-set evaluation uses two modes:

PredCls (Predicate Classification). Ground-truth object labels from the annotations are used directly as the video-level object set $\mathcal{O}_v$. This isolates the relationship prediction ability of the VLM.

SGDet (Scene Graph Detection). Objects are *estimated* from the video content using the VLM itself. The estimation proceeds as follows:

1. **Caption-guided estimation.** The VLM is prompted with all available captions plus the canonical AG object vocabulary (36 foreground classes, including person) as a candidate list and instructs it to exclude person. It returns a JSON list of object name strings.
2. **Vocabulary filtering.** The estimated object set is intersected with the canonical AG vocabulary to ensure all labels conform to the closed label space: $\mathcal{O}_{\text{est}} = \hat{\mathcal{O}} \cap \mathcal{V}$.
3. **Discriminative verification.** For each estimated object $o \in \mathcal{O}_{\text{est}}$, a binary Yes/No verification query is sent to the VLM: "*Is there a $\{o\}$ in this scene? Answer only Yes or No.*" Using batched single-token inference with log-probability extraction, a confidence score $p_{\text{yes}}(o)$ is obtained for every object.

Evaluation Protocol.

Matching Procedure. For each frame and each object, predictions are matched against ground truth by exact object name matching. For objects present in the GT but absent from the VLM predictions (or vice versa), the corresponding entry is recorded as a miss (false negative) or false positive, respectively.

Metrics. For each label l in the vocabulary of an axis (each of the 3 attention labels, 17 contacting labels, or 6 spatial labels):

$$\text{Precision}(l) = \frac{|\text{TP}(l)|}{|\text{TP}(l)| + |\text{FP}(l)|} \tag{71}$$

$$\text{Recall}(l) = \frac{|\text{TP}(l)|}{|\text{TP}(l)| + |\text{FN}(l)|} \tag{72}$$

$$\text{F1}(l) = 2 \cdot \frac{\text{Precision}(l) \cdot \text{Recall}(l)}{\text{Precision}(l) + \text{Recall}(l)} \tag{73}$$

Table 24: **Reported configuration of the zero-shot U-WSGG prompting methods.** K: annotated target frames. Query counts for each stage are listed separately in Table 25.

Dimension	Caption Context (U-WSGG-SUB)	Graph RAG (UWSGG-GRAPHRAG)
Relationship text prefix	Caption transcript	Top-1 node: entities, actions, scenes; no captions
Per-frame visual input	Q_f: target + up to 15 annotated context frames	Same Q_f
Caption proximity sort	Generation prefix	No caption prefix in generation
Shared offline build	Keeps captions (P1)	Uses captions and graph (P1, P0)
Object-specific retrieval	None	BGE embeddings; top-20 candidates, top-1 context
Retrieval reuse	Not applicable	Once per object across K frames
Embedding cache	None	Once per video
Node relevance check (P3)	None	Verification-clip selection only; no effect on reported predictions
Relationship generation (P4)	One prompt per (frame, object)	One prompt per (frame, object)
Label verification (P5)	Optional; disabled	Optional; disabled
SGDet discovery (P6)	Whole-video tensor + captions	Whole-video tensor + captions

Macro-averaged and micro-averaged F1. Macro-F1 averages the per-label F1 scores, giving each label equal weight. Micro-F1 pools true positives, false positives, and false negatives across labels before computing precision, recall, and F1, so frequent labels contribute more.

Attention accuracy. Since attention is single-label, we additionally report top-1 accuracy.

Verification-aware metrics. When verification scores are available (p_{yes}), we evaluate at multiple confidence thresholds $\tau \in \{0.3, 0.5, 0.7, 0.9\}$ to generate precision–recall trade-off curves. A predicted label is accepted only if $p_{\text{yes}} \geq \tau$.

H.3.2 METHOD COMPARISON

Table 24 compares the configuration used in the reported runs. Both methods receive the same target-frame-plus-context tensor Q_f; only the text prefix of the relationship query differs. Label verification is optional and disabled in these runs. Prompt identifiers P0–P5 refer to the pipeline stages of Figure 4 in the main paper: graph extraction (P0), subtitle generation (P1), keyword extraction (P2), node verification (P3), relationship query (P4), and label verification (P5); object discovery is P6. Node relevance checking (P3) selects an optional verification clip, not the graph context used for prediction.

H.3.3 EFFICIENCY AND COMPLEXITY ANALYSIS

Table 25 counts VLM prompts, each producing one response, for a video with K annotated target frames and $M = |\mathcal{O}_v|$ objects. Let K_s be the number of segments in the cached Stage-1 build ($K_s = K$ when the build and evaluation use the same key-frame list), and $N_c = \sum_o n_o \leq MN_r$ the number of object–node checks, where n_o counts retrieved nodes with an available clip for object o and retrieval retains at most $N_r = 20$ nodes. Each node-check prompt contains both Yes/No sub-questions; it counts once.

Deduplication Savings. The VLM overhead of **UWSGG-GRAPHRAG** over U-WSGG-SUB for Steps 1–3 is $M + N_c \leq M + MN_r$ prompts, independent of K because these steps run once per object. For $K = 30$, $M = 8$, and $N_r = 20$ with all 20 nodes retrieved for every object, repeating keyword extraction and node checking at every frame would require $K(M + MN_r) = 5{,}040$ prompts. Deduplication reduces the same two prompt types to $M + MN_r = 168$, exactly $30\times$

Table 25: **VLM prompt counts by stage for the reported configuration.** Offline construction is shared and counted once per cache build. Counts exclude graph-parse retries. SGDet discovery includes one object-list prompt and one confidence prompt per discovered object. Label verification is disabled.

Stage	U-WSGG-SUB	UWSGG-GRAPHRAG
Offline captions and graphs (P1, P0)	$2K_s$	$2K_s$
Object discovery and confidence (SGDet only)	$1 + M$	$1 + M$
Keyword extraction (P2)	0	M
Embedding retrieval (no VLM)	0	0
Node relevance checks (P3)	0	$N_c \leq MN_r$
Relationship answers (P4)	KM	KM
Label verification (P5)	0	0
Online total, PredCls	KM	$M + N_c + KM$
Online total, SGDet	$1 + M + KM$	$1 + 2M + N_c + KM$

fewer. This is a reduction in retrieval-stage prompt counts; the KM relationship-answer prompts remain.

Embedding Cost. Step 2 requires no VLM calls. The BGE model embeds the keyword sets, and similarity calculations compare them with cached graph embeddings. This separates retrieval computation from VLM inference; the query counts alone do not quantify their relative runtime.

Sub-Batching. Relationship and node-check prompts are submitted in chunks of at most 64. In online PredCls, U-WSGG-SUB therefore uses $\lceil KM/64 \rceil$ relationship-generation chunks. **UWSGG-GRAPHRAG** adds one keyword-extraction batch of M prompts and $\lceil N_c/64 \rceil$ node-check chunks. SGDet adds one object-list request followed by one object-confidence batch of M prompts to either method. The fast offline build first submits all caption prompts and then all graph prompts; failed batches are bisected or retried individually. Thus the full pipeline does not have a fixed count of four batched generation calls. These are application-level submissions; scheduling inside vLLM is separate. The BGE embedding model ($\sim$670 MB) shares the GPU with the VLM and its inference buffers.

H.3.4 IMPLEMENTATION DETAILS

All VLMs are deployed via the vLLM inference engine with tensor parallelism. Key performance optimizations include: (i) 8$\times$ frame subsampling for videos with $>$120 frames, (ii) query deduplication (**UWSGG-GRAPHRAG** only), processing unique prompts across steps 1–3 for a $K\times$ reduction, (iii) precomputed graph embeddings shared across queries (**UWSGG-GRAPHRAG** only), (iv) annotation-driven clips matching the graph-construction segmentation, and (v) chunks of 64 for relationship and node-check prompts.

Efficiency accounting. The counts above describe prompts rather than wall-clock runtime. A node check has two question items in one prompt, so N_c node-check prompts contain $2N_c$ sub-questions; the sub-questions are not separate generation requests. Node checking is counted as executed work even though its output only selects an optional verification clip and does not affect the reported predictions. Enabling label verification would add one prompt per predicted label, $KM\,|\overline{\hat{R}}|$, to either method; it contributes zero in the reported runs. Token lengths, graph-parse retries, embeddings and video loading also affect cost. These analytic counts do not supply a measured cross-method latency or peak-memory comparison.

H.3.5 UNLOCALIZED BACKBONE EVALUATION: PROTOCOL AND COMPARABILITY

Main-paper Table 3 compares caption context and Graph RAG across five open-weight backbones in SGDet and PredCls. It uses the ground-truth-plus-corrections annotation basis and reports per-axis precision/recall/F1 and micro-/macro-F1. The object-set matching and metric definitions appear in Appendix H.3.1. These results retain their original evaluated inventories; they are separate from

the R@K evaluation on the test split and from the fixed 150-video matched analysis. A reconciled per-model video manifest is unavailable for the P/R/F1 tables, so they are not treated as full test-split estimates.

H.4 Zero-Shot WSGG Experiments

The zero-shot WSGG methods predict relationships with explicit geometric context. Unlike the zero-shot U-WSGG methods, zero-shot WSGG SGDet also produces 3D boxes, which can be evaluated by oriented 3D IoU.

H.4.1 Visual inputs and the tool loop

Method A makes one structured-generation call per annotated frame. Its input contains a target frame marked with object identifiers, unmarked context frames, a marked bird's-eye view, and a text table of metric OBBs. PredCls supplies GT object identifiers and boxes. SGDet instead uses Grounding DINO proposals with lifted OBBs; the MLLM can retain or add objects and returns boxes and predicates. This is geometry-conditioned prompting, not inference from raw RGB without geometric preprocessing.

Method B follows a fixed sequence controlled by the inference procedure: perceive, retrieve context, propose relationships, verify, repair, and emit. It uses Method A's geometric inputs and adds descriptions retrieved from nearby video clips. The first VLM call produces the graph proposal. A geometric/schema critic returns violations, and a second call repairs the graph when violations occur. The order of these stages is fixed; the model does not choose which tool to call. The no-critic result scores the saved first proposal from the same run; the full result scores the post-repair output. Thus the comparison controls the initial proposal and measures the critic-plus-repair stage, including its extra call.

The critic checks vocabulary/schema validity, floor penetration, implausible extent or size, contact/support distances, and above/beneath consistency. Its contact tests use distances between axis-aligned envelopes of the predicted boxes. It does not check front/behind/side using person orientation, which is unavailable to that check. Reported 3D-IoU matching remains a separate evaluation of oriented boxes.

Summed over the per-frame violation lists of all 1,511 videos in the saved outputs, the violations flagged by the critic fall from 72,807 to 56,463 in PredCls (-22.4%) and from 90,853 to 57,181 in SGDet (-37.1%) after repair. In PredCls, 45,712 of the 72,807 violations concern floor, size, or extent properties of the supplied ground-truth boxes, which the repair call cannot change; only the contact, vertical, and schema checks fall substantially.

In SGDet, the repair call changes 113,194 boxes, of which 35,481 have a same-class ground-truth box. Their mean oriented 3D IoU falls from 0.096 to 0.072, and the number reaching IoU ≥ 0.15 falls from 7,814 to 5,579. Repair therefore satisfies more geometric constraints while moving the changed boxes further from the annotated geometry.

H.5 Zero-shot WSGG SGDet operating points and visibility breakdown

Tables 26–28 extend the zero-shot WSGG SGDet results of both methods (Qwen3-VL-8B) on all 1,511 test videos to $K = 10, 20, 50$ and to the OO and UO buckets, with one table per matching rule: class-only, oriented 3D IoU ≥ 0.15, and 3D IoU ≥ 0.25. They are read from the saved score files of the reported runs, and all values are percentages. The zero-shot scorer keeps one object representation per annotated object, so its GT set contains 773,236 triplets (358,793 UO). Supervised WSGG PredCls and SGDet are both scored on the same 481,790 triplets (67,347 UO), which exclude the objects without a pre-extracted detector feature. The zero-shot bucket values are therefore not comparable with the supervised WSGG bucket values. Per-predicate support for the zero-shot GT layout is not stored in the saved score files.

Table 26: **Zero-shot WSGG SGDet by visibility bucket and K: class-only matching** (no 3D-IoU test).

Metric	Method / Configuration	All Relations			Observed Objects (OO)			Unobserved Objects (UO)		
		@10	@20	@50	@10	@20	@50	@10	@20	@50
WC R	Marked frames + BEV (A)	18.7	23.6	24.4	21.1	26.6	27.5	13.7	17.8	18.7
	Tool loop + critic (B)	**20.6**	**26.7**	**27.5**	**23.9**	**30.8**	**31.6**	**14.0**	**19.1**	**20.1**
	B w/o critic	19.9	25.5	26.4	22.9	29.2	30.0	13.8	18.8	19.7
WC mR	Marked frames + BEV (A)	6.5	12.2	14.0	7.9	13.9	15.7	3.7	7.6	9.2
	Tool loop + critic (B)	**7.5**	**16.8**	**18.5**	**9.2**	**19.4**	**21.0**	**4.0**	**9.4**	**10.9**
	B w/o critic	**7.5**	16.0	17.6	9.1	18.5	20.0	**4.0**	9.3	10.8
NC R	Marked frames + BEV (A)	14.4	25.2	33.3	17.4	29.7	38.2	8.8	16.7	**23.4**
	Tool loop + critic (B)	**15.5**	**27.2**	**33.5**	**19.2**	**32.4**	**38.7**	**9.1**	**17.6**	**23.4**
	B w/o critic	**15.5**	27.0	33.4	**19.2**	32.2	38.6	9.0	17.4	23.3
NC mR	Marked frames + BEV (A)	9.9	15.1	20.2	11.8	17.6	23.2	5.6	9.1	13.7
	Tool loop + critic (B)	12.9	**18.9**	**23.8**	15.4	**22.1**	**27.2**	**6.5**	**10.3**	**14.6**
	B w/o critic	**13.0**	18.5	23.2	**15.6**	21.7	26.5	**6.5**	10.2	**14.6**

Table 27: **Zero-shot WSGG SGDet by visibility bucket and K: 3D IoU ≥ 0.15 matching.**

Metric	Method / Configuration	All Relations			Observed Objects (OO)			Unobserved Objects (UO)		
		@10	@20	@50	@10	@20	@50	@10	@20	@50
WC R	Marked frames + BEV (A)	5.7	7.0	7.2	8.2	10.2	10.5	1.3	1.7	1.8
	Tool loop + critic (B)	**6.4**	**8.1**	**8.3**	**9.4**	**12.0**	**12.3**	**1.4**	**1.9**	**2.0**
	B w/o critic	6.3	8.0	8.2	9.2	11.7	12.0	**1.4**	**1.9**	**2.0**
WC mR	Marked frames + BEV (A)	2.0	3.8	4.3	3.0	5.3	5.9	**0.4**	0.8	1.1
	Tool loop + critic (B)	**2.3**	**5.4**	**5.9**	**3.4**	**7.3**	**8.0**	**0.4**	**1.1**	**1.4**
	B w/o critic	**2.3**	5.3	**5.9**	**3.4**	**7.3**	7.9	**0.4**	**1.1**	**1.4**
NC R	Marked frames + BEV (A)	5.0	8.5	11.0	7.6	12.6	16.0	**0.6**	**1.5**	2.4
	Tool loop + critic (B)	5.0	8.8	10.7	7.7	13.1	15.7	0.5	**1.5**	2.4
	B w/o critic	**5.3**	**9.2**	**11.2**	**8.3**	**13.7**	**16.4**	0.5	**1.5**	**2.5**
NC mR	Marked frames + BEV (A)	3.4	5.0	6.6	5.0	7.2	9.4	**0.5**	**1.0**	1.8
	Tool loop + critic (B)	4.1	6.1	7.7	5.9	8.6	10.5	0.4	**1.0**	**1.9**
	B w/o critic	**4.6**	**6.4**	**7.8**	**6.6**	**9.0**	**10.7**	**0.5**	**1.0**	**1.9**

Table 28: **Zero-shot WSGG SGDet by visibility bucket and K: 3D IoU ≥ 0.25 matching.**

Metric	Method / Configuration	All Relations			Observed Objects (OO)			Unobserved Objects (UO)		
		@10	@20	@50	@10	@20	@50	@10	@20	@50
WC R	Marked frames + BEV (A)	4.0	5.0	5.2	5.9	7.4	7.6	**0.9**	1.1	1.2
	Tool loop + critic (B)	**4.6**	**5.9**	**6.0**	**6.7**	**8.6**	**8.8**	**0.9**	**1.2**	**1.3**
	B w/o critic	4.5	5.7	5.9	6.6	8.4	8.6	**0.9**	**1.2**	**1.3**
WC mR	Marked frames + BEV (A)	1.5	2.7	3.1	2.2	3.9	4.4	0.2	0.4	0.5
	Tool loop + critic (B)	**1.7**	**3.9**	**4.3**	**2.5**	**5.4**	**5.9**	**0.3**	**0.6**	**0.7**
	B w/o critic	**1.7**	3.8	**4.3**	**2.5**	5.3	**5.9**	**0.3**	**0.6**	**0.7**
NC R	Marked frames + BEV (A)	3.6	6.1	7.9	5.5	9.1	11.6	**0.4**	0.9	1.5
	Tool loop + critic (B)	3.6	6.3	7.7	5.6	9.5	11.4	0.3	0.9	**1.6**
	B w/o critic	**3.9**	**6.6**	**8.0**	**6.0**	**10.0**	**11.8**	0.3	**1.0**	**1.6**
NC mR	Marked frames + BEV (A)	2.5	3.5	4.7	3.7	5.2	6.9	**0.2**	0.4	0.9
	Tool loop + critic (B)	3.0	4.3	5.5	4.4	6.3	7.8	**0.2**	**0.5**	0.9
	B w/o critic	**3.4**	**4.6**	**5.6**	**4.9**	**6.6**	**7.9**	**0.2**	**0.5**	**1.0**

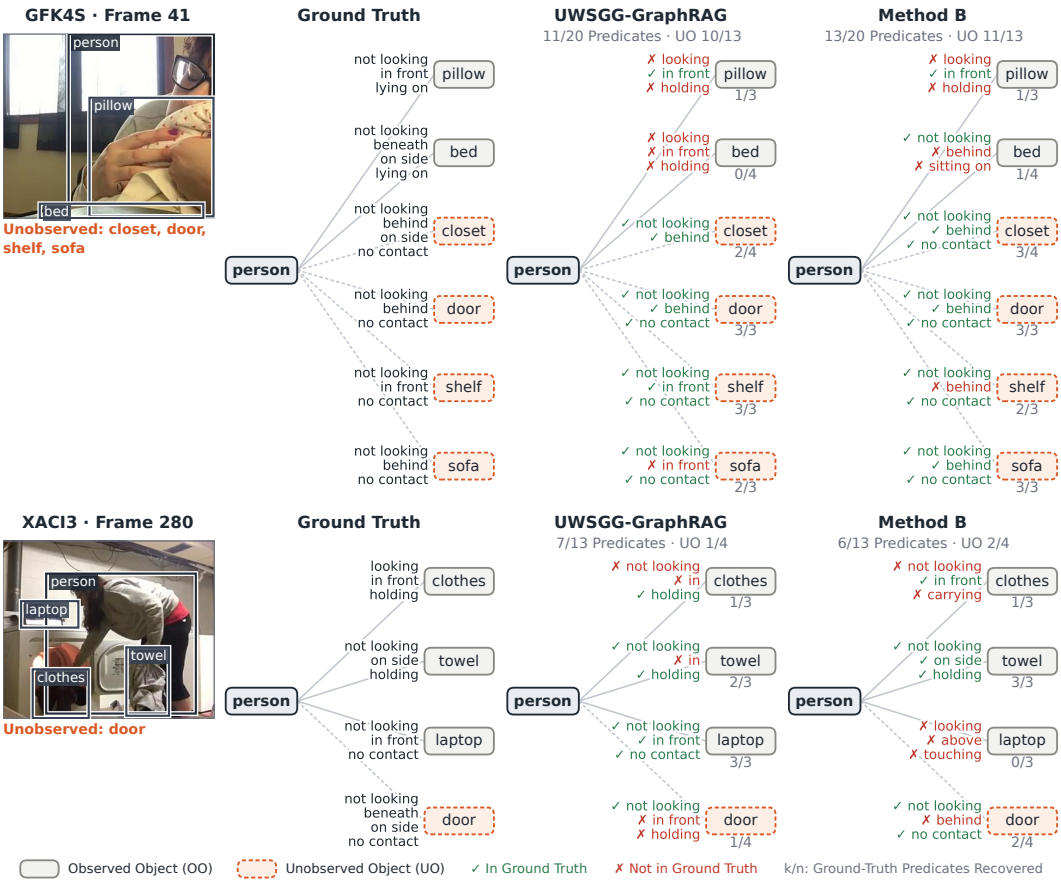


Figure 51: **Zero-shot PredCls scene graphs on two test frames**, predicted by **UWSGG-GRAPHRAG** (zero-shot U-WSGG) and Method B (zero-shot WSGG), in the layout of Figure 38. Top (GFK4S): the four unobserved objects have no pre-extracted features, so the supervised WSGG models make no prediction for them, whereas both zero-shot methods predict every edge; Method B recovers 11 of their 13 ground-truth predicates and **UWSGG-GRAPHRAG** 10. Bottom (XACI3, the top frame of Figure 38): **UWSGG-GRAPHRAG** recovers 7 and Method B 6 of the 13 ground-truth predicates, against 12 for WorldWise++.

H.6 QUALITATIVE ZERO-SHOT SCENE GRAPHS

Figure 51 shows PredCls scene graphs of **UWSGG-GRAPHRAG** and Method B, which predict an edge for every annotated object. On GFK4S, four of the six objects (the closet, door, shelf, and sofa) are out of view and have no pre-extracted features, so the supervised WSGG models cannot predict their edges, whereas both zero-shot methods do, mostly correctly. On XACI3, where every object has a pre-extracted feature, both zero-shot methods recover about half of the ground-truth predicates, fewer than WorldWise++ (Figure 38).

H.7 EXTERNAL MLLM BASELINES AND CONTEXT CONTROLS

We compare the zero-shot methods with two recent approaches under the scorers used above. SceneGraphVLM Makarov et al. (2026) generates a scene graph with 2D boxes for each frame, conditioned on its own graph for the preceding frame, and is trained on the Action Genome training split with rewards that penalize hallucinated objects and relations. We run its released Action Genome checkpoint without retraining; unlike the zero-shot methods, it is therefore supervised in-domain. SGR3 Wang et al. (2026c) retrieves reference scene graphs with a visual late-interaction retriever to guide an MLLM; its code is not released, so we reimplement its retrieval inside **UWSGG-GRAPHRAG** and hold the MLLM, the visual input, the object inventory, and the context budget fixed.

SceneGraphVLM. Table 29 reports class-only SGDet on the full split. SceneGraphVLM has the highest observed-object recall of all MLLM rows (OO NC R@50 of 42.2 against 38.7 for Method B) but recovers few unobserved relationships (UO NC R@50 of 8.0 against 23.4), so Method B retains the higher WC R@20 (26.7 against 22.3). It also predicts far fewer unsupported objects, whose class is absent from the object inventory of the video (28.1% of its predicted objects against 55.3–74.4% for the zero-shot SGDet methods), and unsupported relations, predicates that are not annotated for their ground-truth pair and frame (20.7% against 47.5–50.5%). The Graph RAG rates cover the 1,480 videos with saved SGDet outputs.

Insights: (1) Supervision on Action Genome labels teaches frame-level relationships but not permanence: those labels contain no relationships for unobserved objects, and a model that conditions only on the preceding frame has no memory of an object once it leaves the view. (2) The lower unsupported rates partly reflect a sparser output rather than better grounding alone, since SceneGraphVLM proposes 1.86 objects per frame against 6.74–11.00 for the zero-shot methods, and its training matches the annotation vocabulary and labelling style. Because the annotations are incomplete, both rates are upper bounds, and we treat the comparison as indicative until a human audit of the flagged predictions is available. (3) The preceding-graph idea transfers to our pipeline without retraining. Adding the prediction for the previous frame to the Qwen3-VL-8B per-object prompt without text context raises WC R@20 from 47.7 to 49.9 on the 150-video subset (Table 30; paired bootstrap over videos, 95% interval of $+1.6$ to $+2.8$), with the gain on observed objects (OO NC mR@50 from 47.6 to 50.8) while unobserved mean recall does not improve (UO NC mR@50 from 54.8 to 54.3).

SGR3-style retrieval. Table 30 reports the retrieval arms on the matched 150-video subset under a fixed budget of $B = 90$ context tokens, the mean length of the **UWSGG-GRAPHRAG** context. SGR3-style retrieval with $k = 3$ reference graphs gives the highest WC R@20 (52.7), compared with 47.4 for the reported **UWSGG-GRAPHRAG** run and 47.8 without context.

Insights: (1) Most of this gain does not come from retrieval. Graphs from a random training video recover most of it (50.9 WC R@20), and retrieval adds 1.8 WC R@20 and 3.8 OO NC mR@50 over that control. Any reference graph exposes the model to the predicate vocabulary: the share of out-of-vocabulary contacting answers falls from 20.2% without context to 3.6% with a retrieved graph and 3.8% with a random one. (2) Retrieval across scenes does not help unobserved objects: no arm exceeds the context-free rows on UO NC mR@50 (54.4 and 54.7 over two sampling draws), because a reference graph from another home carries no information about the objects of the queried video. (3) The within-video retrieval of **UWSGG-GRAPHRAG** does not improve over the context-free prompt with Qwen3-VL-8B either (WC mR@20 of 28.8 against 30.6), since the top-1 node changes little with the queried object (2.3 distinct contexts per frame for 4.4 objects per video). Together with the backbone dependence discussed in Appendix H.3.5, this indicates that the gains attributed to retrieval are specific to the backbone and to the prompt format.

Table 29: **SceneGraphVLM versus the zero-shot WSGG methods under class-only SGDet matching.** Rows A and B repeat Table 26. *Released checkpoint (Qwen3.5-0.8B) trained on the Action Genome training split and run without retraining, so it is supervised in-domain.

Metric	Method	All Relations			Observed Objects (OO)			Unobserved Objects (UO)		
		@10	@20	@50	@10	@20	@50	@10	@20	@50
WC R	Marked frames + BEV (A)	18.7	23.6	24.4	21.1	26.6	27.5	13.7	17.8	18.7
	Tool loop + critic (B)	20.6	**26.7**	**27.5**	23.9	30.8	31.6	**14.0**	**19.1**	**20.1**
	SceneGraphVLM*	**22.3**	22.3	22.3	**33.1**	**33.2**	**33.2**	5.5	5.5	5.5
WC mR	Marked frames + BEV (A)	6.5	12.2	14.0	7.9	13.9	15.7	3.7	7.6	9.2
	Tool loop + critic (B)	7.5	16.8	**18.5**	9.2	19.4	21.0	4.0	**9.4**	**10.9**
	SceneGraphVLM*	**16.9**	**16.9**	16.9	**22.1**	**22.1**	**22.1**	**6.1**	6.1	6.1
NC R	Marked frames + BEV (A)	14.4	25.2	33.3	17.4	29.7	38.2	8.8	16.7	**23.4**
	Tool loop + critic (B)	15.5	**27.2**	**33.5**	19.2	32.4	38.7	**9.1**	**17.6**	**23.4**
	SceneGraphVLM*	**24.3**	25.6	28.7	**36.3**	**38.3**	**42.2**	5.8	6.2	8.0
NC mR	Marked frames + BEV (A)	9.9	15.1	20.2	11.8	17.6	23.2	5.6	9.1	13.7
	Tool loop + critic (B)	12.9	18.9	23.8	15.4	22.1	27.2	6.5	**10.3**	**14.6**
	SceneGraphVLM*	**22.2**	**27.6**	**33.1**	**29.5**	**35.6**	**42.7**	**7.1**	8.0	12.3

Table 30: **Context ablations for zero-shot U-WSGG PredCls on the matched 150-video subset**: Qwen3-VL-8B, standard decoding, the same object inventory, visual input and answer budget in every row; only the text context changes. SGR3-style rows reimplement the visual retrieval of SGR3 Wang et al. (2026c) (SigLIP2 patch embeddings, late-interaction scoring) over a bank of 7,516 ACTIONGENOME4D training videos that shares no video with either test set, and insert the top-k reference graphs; the random control inserts the graph of a random training video. [a]Graph RAG prompt with an empty context. [b]Reported run, whose context exceeds B (maximum 207 tokens). [c]Per-object prompt without text context; the preceding-frame graph is the no-context prediction for the previous frame. Tokens: mean context tokens per prompt (Qwen3-VL tokenizer, header included). Markers rank rows within each block. Values are percentages (%).

Context	All Relations				OO	UO		Tokens
	WC R@20	WC mR@20	NC R@50	NC mR@50	NC mR@50	NC R@50	NC mR@50	
Retrieved context *(budget $B = 90$ tokens)*								
None[a]	47.8	30.6	60.6	48.8	47.3	53.2	54.4	0
None[a], second draw	47.8	**30.7**	60.6	49.3	47.7	53.3	**54.7**	0
Graph RAG top-1 node, uncapped[b]	47.4	28.8	60.0	48.5	46.1	53.9	53.4	90.3
Graph RAG top-1 node	47.2	29.0	60.0	48.6	46.2	53.5	53.6	74.2
SGR3-style, k=1	51.8	29.9	65.3	49.5	48.3	60.0	53.7	82.1
SGR3-style, k=3	**52.7**	30.6	**66.1**	**50.5**	**49.7**	**60.9**	54.2	89.2
SGR3-style, k=5	52.1	29.1	63.5	48.0	46.3	59.0	52.6	89.3
Graph RAG ($B/2$) + SGR3-style, k=1	50.9	29.8	63.1	49.1	47.3	57.7	54.0	87.7
Random training graph	50.9	29.4	64.4	48.5	45.9	60.0	54.3	80.3
Preceding-graph context								
None[c]	47.7	30.5	60.5	49.2	47.6	53.1	**54.8**	–
Preceding-frame graph[c]	**49.9**	**30.9**	**62.7**	**50.7**	**50.8**	**57.0**	54.3	–

I Extended Context and Related Work

I.1 Extended Literature on Structured Scene Understanding

To complement the taxonomic overview in Section 2 (Table 1), this section provides an extended review of structured scene graphs across 2D static images, temporal video sequences, weakly-supervised formulations, 3D/4D spatial domains, and downstream embodied applications.

Static image scene graphs. Early foundations established scene graphs as graph-structured semantic abstractions linking localized object entities via directional relational predicates Krishna et al. (2016); Chen et al. (2019); Zhu et al. (2022). While early frameworks decoupled object proposal generation from relation classification, subsequent advances integrated these stages into end-to-end architectures; for instance, SGTR Li et al. (2022a) utilizes bipartite matching with Transformers to predict entity nodes and relationship edges concurrently. Concurrently, learning scene graphs under natural language supervision Zhong et al. (2021) relaxed the requirement for dense pairwise manual labels by aligning visual relation representations with textual descriptions, and environment-invariant curriculum training Min et al. (2023) introduced progressive schedules that stabilize learning on difficult relational tails.

Temporal video scene graphs. Transitioning from static frames to video necessitates tracking identity continuity and capturing time-varying interactions. Early benchmarks such as Vid-VRD Shang et al. (2017) framed video relation detection around tracklet pairs, while ActionGenome (AG) Ji et al. (2019) grounded dense human–object interaction graphs across sequential video frames. Subsequent methodologies developed specialized architectures for dynamic relational modeling: STTran Cong et al. (2021) introduced factorized spatial and temporal Transformer attention; memory-augmented models Feng et al. (2022) integrated recurrent or slot-based state memories to bridge long temporal gaps; Tempura Nag et al. (2023) tackled severe class imbalance across predicate categories; and ImparTail Peddi et al. (2025) established robust, unbiased prediction under tail distributions. Expanding beyond two-stage tracking-and-prediction, OED Wang et al. (2024a) proposed a fully one-stage framework; UNO Le et al. (2026) consolidated one-stage video SGG through object-centric representation learning; HyperGLM Nguyen et al. (2025a) leveraged hypergraph message passing to model high-order multi-entity correlations; and action scene graphs Rodin et al. (2023) enable long-form understanding of egocentric activities. Furthermore, panoptic VidSGG Yang et al. (2023) incorporated background stuff regions alongside foreground things, while SceneSayer Peddi et al. (2024) established the task of scene graph anticipation (SGA) to forecast future relational topologies ahead of time.

Weakly-supervised and open-vocabulary dynamic SGG. To alleviate the heavy annotation burden of frame-level relational graphs, recent efforts explore weaker supervisory signals and open-vocabulary transfer. Weak supervision strategies exploit narrated text and language transcripts Kim et al. (2025), while TRKT Xu et al. (2025) transfers relation-aware knowledge across temporal intervals. LLMs have also been leveraged to guide pseudo-relation supervision Kim et al. (2024), distill composite visual cues for zero-shot relation prediction Li et al. (2023), and drive zero-shot local graph generation via foundation model priors Zhao & Xu (2023). In panoptic contexts, VL-Prompt Zhou et al. (2024) employs vision-language prompt tuning. In parallel, architecture-centric refinements include embedding structured spatio-temporal knowledge graphs into Transformers (SKET) Pu et al. (2023), direct graph extraction via Transformer decoders (EGTR) Im et al. (2024), pose-aware visibility matrix modulation (SpGLT) Wu et al. (2025b), implicit language reasoning in LLM decoders (SceneLLM) Zhang et al. (2025b), and joint denoising with debiasing for dynamic graphs (TD^2-Net) Lin et al. (2024).

3D and 4D scene graph representations. Moving beyond 2D pixel coordinates, 3D scene graphs contextualize entities and interactions within a global metric reference frame, originally proposed as compact semantic environment representations for mobile robotics Kim et al. (2020). Multimodal spatial knowledge accumulation frameworks Feng et al. (2023) advanced 3D relation reasoning over point cloud reconstructions. Subsequent developments explored self-supervised representation learning via object-level 3D reconstruction (SGRec3D) Koch et al. (2023), open-vocabulary semantic parsing in indoor functional environments Zhang et al. (2025a), and outdoor hierarchical graph

mapping across large physical scenes Nyffeler et al. (2025). For robust inference, statistical confidence rescoring Yeo et al. (2025) counters point sparsity in multi-view 3D graphs, while FROSS Hou et al. (2025a) achieves online, real-time 3D SGG from streaming RGB-D sensors. Expanding into the temporal continuum, 4D panoptic scene graph formulations Yang et al. (2024) integrate semantic segments and temporal tracking in 3D, rich 2D scene lifting techniques Wu et al. (2025d) train 4D graphs from abundant video data, and RealGraph Lin et al. (2023) provides multi-view dynamic context benchmarks.

Downstream embodied and generative applications. Scene graphs provide structured scene representations that power diverse downstream applications. In visual synthesis, relational graphs guide controllable semantic image manipulation Dhamo et al. (2020), meta-image generation (MIGS) Farshad et al. (2021), unified image-to-graph translation (Relationformer) Shit et al. (2022), and graph-conditioned video synthesis (SSGVS) Cong et al. (2022). In conversational and long-video reasoning, dynamic graph representations facilitate multi-turn video dialogue Geng et al. (2021) and hierarchical video parsing via interlaced graph levels (HIG) Nguyen et al. (2024). For robotics and embodied AI, Scene Graph Memory Kurenkov et al. (2023) maintains persistent world representations in changing environments, interactive manipulation studies ground spatial and contact predicates from physical experience Migimatsu & Bohg (2022), and ESCA Huang et al. (2025a) contextually grounds embodied agent navigation through ongoing scene graph generation.

I.2 Extended Literature on Vision-Language Models for Spatial Reasoning

While Section 2 introduces the role of Vision-Language Models (VLMs) and Multi-Modal Large Language Models (MLLMs) in scene reasoning, this section surveys their recent spatial adaptations in greater depth, focusing on 2D depth-grounding, spatial probing benchmarks, and explicit 3D architectural formulations.

Depth-grounded and spatial prompting in VLMs. Standard vision-language pretraining primarily emphasizes semantic categorization and high-level image captioning, often leaving models deficient in geometric and relational precision. To bridge this gap, SpatialVLM Chen et al. (2024) synthesized internet-scale spatial question-answering pairs to instill quantitative metric understanding. Complementary lines of work integrate metric cues directly into token representations: SpatialRGPT Cheng et al. (2024) incorporates depth-aware region proposals; SpatialBot Cai et al. (2025) blends high-resolution RGB observations with explicit depth sensor streams; and SD-VLM Chen et al. (2025b) designs dual-stream encoders for direct spatial distance estimation. In parallel, training-free prompting strategies such as See&Trek Li et al. (2025b) steer multimodal models toward spatial coordination without weight updates. Other works enhance spatial reasoning via targeted post-training or architectural enhancements: fine-grained preference optimization Shen et al. (2026) aligns relational answers with geometric ground truths, mental imagery simulation Lee et al. (2025) equips models with perspective-aware transformations, Struct2D Zhu et al. (2025a) guides spatial deduction through perceptual token hierarchies, empirical probes by Wen et al. Wen et al. (2024) evaluate the intrinsic ability of Transformer attention to capture spatial relationships, and SpatialMLLM Wu et al. (2025a) injects visual spatial priors directly into multimodal decoders.

Spatial intelligence evaluation benchmarks. A concurrent series of comprehensive benchmarks has been established to diagnose whether current foundation models can faithfully interpret spatial-temporal physical worlds. Specifically, SITE Wang et al. (2025a) assesses fine-grained spatial intelligence across multi-view consistency and physical containment; STI-Bench Li et al. (2025d) evaluates precise spatial reasoning and metric geometric queries in 3D; and CoSpace Zhu et al. (2025b) benchmarks spatial understanding over continuous coordinate domains, systematically highlighting the limitations of conventional text-generation objectives on geometric localization tasks.

Explicit 3D reasoning and spatial world models. Moving beyond 2D projection and depth prompting, recent architectures explicitly incorporate 3D geometry into language models. SpatialLLM Ma et al. (2025b) introduces 3D-informed token structures that fuse point clouds with language prompts; SpatialReasoner Ma et al. (2025a) generates chain-of-thought rationales grounded in explicit 3D coordinate frames; SpatialLM Mao et al. (2025) enables language models to generate structured 3D indoor scene layouts directly; and scalable 2D-to-3D lifting frameworks Miao et al.

(2025) synthesize large-scale 3D training signals from existing 2D video repositories. In layout synthesis, LayoutVLM Sun et al. (2025) formulates 3D bounding box prediction as a differentiable optimization problem conditioned on language semantics. Furthermore, test-time scaling techniques such as MindJourney Yang et al. (2025) leverage generative world models to search through simulated spatial rollouts during inference, while Zheng et al. Zheng et al. (2025) enhance MLLM reasoning by pretraining on video sequences augmented with 3D vision-geometry priors.

I.3 Neural Memory Models

Modern Hopfield networks and dense associative memories offer exponential storage capacity Lucibello & Mézard (2024); Hu et al. (2024); Agliari & Marzo (2020); Clark (2025) and differentiable retrieval dynamics, making them attractive for architectures requiring persistent state. The Hopfield framework has been extended to long sequences as attention-compatible memory layers Chaudhry et al. (2023), while biological extensions Kozachkov et al. (2025) and connections to diffusion models Hoover et al. (2024b;a) have broadened the theoretical foundations. These mechanisms have been integrated into large-scale models: CAMELoT He et al. (2024) serves as a training-free associative memory for frozen LLMs, M+ Wang et al. (2025d) scales knowledge retention via long-term memory, and CLOOB Fürst et al. (2022) uses associative retrieval for contrastive vision-language pretraining.

Prototypical networks. Prototypical networks Snell et al. (2017) learn a metric space and classify examples by comparing them with a prototype representation of each class. This provides a framework for few-shot learning. This paradigm has been extended in several directions: Graph Prototypical Networks Ding et al. (2020) adapt prototypical representations to attributed networks, enabling few-shot node classification by constructing a pool of semi-supervised tasks that mimic the test environment; ProFusion Zhao et al. (2025) augments prototypical networks with multimodal feature fusion, constructing image, text, and fused prototypes from vision-language pre-trained models for robust few-shot classification; prototype memory Smirnov et al. (2022) scales prototypical representations for large-scale face recognition; and training-free prototype calibration Wang et al. (2023) addresses few-shot class-incremental learning by adjusting prototypes without retraining. In the context of scene graph generation, RA-SGG Yoon et al. (2024) proposes retrieval-augmented SGG via multi-prototype learning, where multiple prototypes per predicate class capture the diversity of visual relationship appearances and mitigate the long-tail distribution problem inherent in relationship annotations.

I.4 Recent Dynamic Scene Graphs and Persistent World State

Recent video scene graph work has improved temporal association, relation prediction, and supervision. VISA Li et al. (2025c) uses memory-enhanced object features and semantic debiasing, whereas SAMJAM Li et al. (2025a) combines segmentation-based tracking with vision-language predictions to stabilize identities in egocentric video. THYME Nguyen et al. (2025b) uses hierarchical context and cyclic refinement across frames, and salient temporal encoding Zhu (2025) selects informative temporal edges rather than connecting every object across frames. UNO Le et al. (2026) unifies box-level and panoptic video SGG with object-centric representations; SVG2/TRaSER Gao et al. (2026) couples a larger panoptic video-graph corpus with trajectory-aware generation. PAWS Kang et al. (2026) studies weakly supervised relation grounding, while FReMuRe Li et al. (2026) and MOSA Wang et al. (2026b) address predicate frequency and motion cues. SceneGraphVLM Makarov et al. (2026) uses preceding-graph context, and OvDSGG Helsby et al. (2026) extends dynamic prediction to open vocabularies. T-STAR Wang et al. (2026a) considers identity-consistent panoptic relations with temporal spans in satellite video. Together, these papers sharpen video-graph prediction, although their reported tasks do not jointly require world-frame 3D boxes and relation states for objects outside the current view.

Scene graphs also serve as structured context for reasoning over long videos. STEP Qiu et al. (2025) induces spatio-temporal graphs to generate compositional question–answer supervision for video LLMs. VideoMindPalace Huang et al. (2025b) organizes hand–object interactions, activity zones, and environment layout in an environment-grounded graph for long-video analysis. SG-Ego/GLEN Pistilli et al. (2026) uses egocentric graph sequences to model activity-conditioned scene changes, while EgoSG Taluzzi et al. (2026) supplies compact temporal graph context to an MLLM.

Semantic Event Graphs Dixit & Liang (2026) prune human–object event histories for long-video question answering, and STSG-VQA Li & Sarikaya (2026) grounds temporal questions in object and event continuity in surgical video. These works show the value of explicit relations for downstream reasoning; WSGG instead evaluates the graph state itself at each timestamp, including relationships involving unobserved objects.

A complementary line of work builds or queries relational 3D scenes. 3DGraphLLM Zemskova & Yudin (2025) encodes semantic 3D object relations for language grounding and question answering. VIZOR Madhavaram et al. (2026) studies viewpoint-invariant spatial relations, GEODE Feng et al. (2026) models geometry and predicates jointly, and ReLaGS Xie et al. (2026) represents relations over language-distilled Gaussian scenes. OP3DSG Kim & Kim (2026) adds parts, functional relations, and affordances; LEXI-SG Kassab et al. (2026) is especially relevant because it derives room-level 3D graph structure and tracks objects from monocular RGB input. ProClosure Muthuraj et al. (2026) also uses monocular video to assign objects to rooms. RAG-3DSG Chang et al. (2026) retrieves object-level evidence to correct uncertain 3D graph semantics, whereas SGR3 Wang et al. (2026c) retrieves scene graphs to aid MLLM relation reasoning. Their retrieval roles connect to our graph-retrieval context mechanism, but serve different prediction and evaluation protocols.

Maintaining a spatial graph as a scene changes is particularly close to the persistence requirement in WSGG. Hi-Dyna Graph Hou et al. (2025b) anchors local dynamic subgraphs to a global layout, Embodied VideoAgent Fan et al. (2025) updates persistent object memory from egocentric video and embodied sensors, and ST-OVSG Wang et al. (2025b) associates open-vocabulary 3D objects over time for robot planning. DeWorldSG Kim et al. (2026) refines RGB-D graph relations using a world model; NoPA Yeo et al. (2026) propagates online object associations and relations; and PUF Yang et al. (2026) fuses uncertain observations into a persistent 3D graph. DSG Liao et al. (2026) updates object relations in changing indoor scenes, while DGSG-Mind Ge et al. (2026) uses dynamic Gaussian scene graphs for longer-term grounding. DAAAM Gorlo et al. (2026) builds a globally consistent hierarchical 4D graph for spatio-temporal queries, R4DSG Ma et al. (2026) maintains anchor-relative changes and object identities from long RGB egocentric video, and GESTO Bartoli et al. (2026) grounds human–object events in persistent 4D memory. Kairos Catalano et al. (2026) adds forecasts of pedestrian presence and flow. These methods make persistent state useful for mapping, planning, or question answering; our target is a per-timestamp world scene graph with localized objects and relations even when some participants are temporarily unobserved.

Object permanence has also been studied through geometric completion rather than scene-graph prediction. Video amodal segmentation Chen et al. (2025a) infers occluded image regions using diffusion priors. From monocular RGB video, 4D Primitive-Mache Mazur et al. (2026) retains 3D primitives through occlusion by extrapolating their motion. PersistGS Ramlal & Zelek (2026) uses physical trajectories to maintain object geometry during complete occlusion in 4D Gaussian reconstruction. These approaches help explain how hidden geometry may persist; WSGG additionally asks which entities and relationships remain in the world state while they cannot be observed.

I.5 Direct Comparison and Evaluation Scope

Video graphs and persistent world graphs. VISA Li et al. (2025c) and OvDSGG Helsby et al. (2026) improve video relation prediction; SAMJAM Li et al. (2025a) stabilizes tracked identities; UNO Le et al. (2026) and SVG2/TRaSER Gao et al. (2026) extend box-level or panoptic video graphs. Their published outputs and benchmarks focus on relations attached to image observations or tracked image regions. Our task retains an object in the world-state inventory when its visibility becomes zero and asks for its world-frame oriented box and current relationships at annotated timestamps. WorldWise++ implements this distinction with persistent object-queries, world-frame geometric inputs, temporal attention, and image-grounded entity decoding. A persistent identity alone does not determine the predicate: an object may remain in the world while its relationship to the observed person changes.

3D mapping and 4D memory. LEXI-SG Kassab et al. (2026) shares our monocular RGB input and tracks objects in a hierarchical 3D map, but evaluates mapping, reconstruction, and segmentation rather than dense person–object predicates during occlusion. R4DSG Ma et al. (2026) is particularly close in its use of long RGB egocentric video and persistent identities; it stores anchor-relative changes for object-centric question answering rather than predicting a globally aligned oriented

box and predicate labels for each annotated person–object pair. DAAAM Gorlo et al. (2026) and GESTO Bartoli et al. (2026) build 4D memories for spatial or activity questions from RGB-D observations, while NoPA Yeo et al. (2026) and PUF Yang et al. (2026) emphasize online fusion and association in 3D maps. Our benchmark supplies world-frame geometry and relationship annotations for the video-level object inventory; our graph model consumes object representations, boxes, and camera poses from the preceding geometric pipeline. It is an offline method with access to the complete video, so its results do not establish online mapping or causal prediction.

Occlusion geometry and hidden-object relations. Video amodal segmentation Chen et al. (2025a) recovers hidden image extent, 4D Primitive-Mache Mazur et al. (2026) carries 3D primitives through monocular-video occlusion, and PersistGS Ramlal & Zelek (2026) uses physics to preserve reconstructed geometry during complete occlusion. These methods address where the hidden object or its shape may be. WSGG also evaluates the predicate connecting that object to the person at the queried timestamp. WorldWise++ keeps the missing-appearance object-query in the relational decoder and predicts attention, spatial, and contacting predicates from temporal context and pair geometry. Its geometric scaffold supplies the hidden object's localization; the relation network does not recover that geometry from an empty visual observation alone.

Graph-conditioned reasoning and retrieval. STEP Qiu et al. (2025), VideoMindPalace Huang et al. (2025b), and EgoSG Taluzzi et al. (2026) use graphs to support video-language reasoning; SG-Ego/GLEN Pistilli et al. (2026) forecasts graph edits from activities. RAG-3DSG Chang et al. (2026) retrieves evidence to repair 3D graph semantics, and SGR3 Wang et al. (2026c) retrieves graphs for MLLM relation reasoning. Our **UWSGG-GRAPHRAG** instead retrieves graph context for per-object predicate queries evaluated against ACTIONGENOME4D annotations. It is a zero-shot approach that prompts an MLLM, distinct from the trained WorldWise++ model; as a zero-shot U-WSGG method, it emits relations without boxes. We evaluate zero-shot WSGG graph generation separately with geometric inputs and box outputs.

Scope of the reported comparison. The general WSGG formulation permits arbitrary object pairs, but the evaluated ACTIONGENOME4D benchmark contains person–object edges at annotated frames, with the person observed; unobserved–unobserved pairs are not scored. The methods use the full video, and WorldWise++ uses bidirectional temporal attention. Supervised WSGG relation scores are conditional on pre-extracted object features, which omit some unobserved objects. Moreover, supervised WSGG SGDet matches visible objects by class and 2D IoU and unobserved objects by class alone, whereas zero-shot WSGG SGDet uses oriented 3D IoU. Consequently, supervised WSGG SGDet recall does not by itself validate 3D box accuracy, and scores across these formulations should not be read as a controlled numerical ranking. Section 5 and Appendix F.3.2 give the corresponding coverage, matching, and architectural details.

J LIMITATIONS

We summarize the limitations of the task formulation, the ACTIONGENOME4D annotations, the proposed methods, and the evaluation, and outline the directions they open for future work.

Scope of the task formulation. ACTIONGENOME4D inherits the actor-centric design of ActionGenome: every annotated relationship connects the observed person to an object, and the videos depict indoor household activities. Although the WSGG formulation permits arbitrary object pairs, object–object relationships, including those between two unobserved objects, are neither annotated nor scored (Appendix A.1). The label space is the closed vocabulary of ActionGenome over the attention, spatial, and contacting axes. Consequently, our results do not establish how the methods behave in scenes with multiple actors, in crowded or outdoor environments, or with open-vocabulary objects and predicates.

Offline world-state completion. We study the offline setting, in which methods observe the complete video. WorldWise++ applies bidirectional temporal attention, so the representation of an object that is unobserved at time t can draw on evidence from later frames. The reported numbers therefore measure offline completion rather than causal prediction from a video prefix; an online variant, which an embodied agent acting in real time would require, remains to be developed and evaluated.

Dependence on monocular reconstruction. The world-frame geometry of ACTIONGENOME4D is derived from monocular reconstruction with π^3, together with Grounding DINO detections and SAM2 masks (Appendices C and D.1). Errors in the reconstructed depth, scale, and camera pose therefore propagate into the oriented 3D boxes. Manual correction reduces these errors (Appendix D.2), but it operates within the reconstructed scene, so agreement among annotators does not rule out a geometric bias shared with the reconstruction.

Annotation provenance. Visible relationships retain the ActionGenome labels, whereas the relationships of unobserved objects are seeded by open-source VLMs. In the training split, these proposals serve as pseudo-labels without human correction; in the test split, annotators correct them (Appendix E.2.2). Supervision for unobserved objects is therefore noisier in training than in evaluation; the adapted baselines train on these pseudo-labels, whereas the reference WorldWise recipe excludes them. The correction logs show substantial editing (65.4% of reviewed spatial labels changed; Table 8), and the annotation agreement reaches 89–92% across the three axes on 200 videos. However, neither statistic measures the residual anchoring toward proposals that annotators accepted unchanged. Moreover, the relationships of an unobserved object are inferred from the surrounding frames rather than read directly from the target frame, which makes them inherently more ambiguous than those of visible objects.

Learning to complete unobserved objects. WorldWise++ learns completion from artificial masks: visible appearances are masked with probability 0.3, and naturally unobserved pairs receive no relationship supervision (Section 5.1). An artificial mask removes the appearance of an object that remains in view, whereas natural unobservability also arises from long absences and from objects leaving the field of view, so the two distributions differ. In addition, we train only the graph model on pre-extracted visual and geometric features, so the relationship objective cannot correct errors in these features.

Recall on unobserved objects. In SGDet, the relationships of an unobserved object can only be recovered if the object has a representation, so proposal misses bound the recall on unobserved objects (UO). Consequently, the raw UO recall of WorldWise++ remains below that of the adapted baselines (R@20 of 14.1 versus 20.0–22.1 in Table 2), even though its UO mean recall is higher (mR@20 of 11.8 versus at most 6.6). This pattern is consistent with the baselines favoring the dominant head predicates, whereas the logit adjustment of WorldWise++ trades part of this head-predicate recall for balanced predicate coverage.

Comparability across evaluation protocols. The reported tables do not form a single controlled ranking. First, supervised WSGG relation scores are conditional on pre-extracted object features, which omit some unobserved objects, whereas the zero-shot methods do not depend on these features. Second, supervised WSGG SGDet matches visible objects by class and 2D IoU and unobserved objects by class alone, whereas zero-shot WSGG SGDet uses oriented 3D IoU and zero-shot U-WSGG matches object sets without boxes; supervised WSGG SGDet recall therefore does not by itself validate 3D box accuracy. Third, the zero-shot P/R/F1 results in Table 3 retain their original

evaluated inventories and are not full test-split estimates (Appendix H.3.5), and the supervised and zero-shot methods differ in supervision, backbone, and inference budget unless explicitly matched. The scope paragraph of Appendix I.5 discusses these differences further.

Future work. We acknowledge that a unified evaluation protocol that scores every method on the complete annotated inventory with common 3D matching, an annotation control initialized without VLM proposals, and an online variant of WSGG are important directions for future work, as is scaling beyond the actor-centric, closed-vocabulary setting of ActionGenome toward downstream embodied AI tasks.