CoMO4D: Temporally Consistent Multi-Object 4D Generation from a Single Monocular Video

Anonymous authors
Paper under double-blind review

Abstract

Observation-conditioned generative reconstruction of temporally consistent multi-object 4D scenes from a monocular video is fundamental to dynamic scene modeling, yet existing methods often struggle to retain semantic object organization and maintain spatial and temporal consistency under occlusion. We propose CoMO4D, a structured generative 4D reconstruction framework that explicitly represents each tracked dynamic rigid object with a single shared canonical geometry and a sequence of frame-wise spatial states, thereby describing both the stable object structure and its time-varying spatial evolution throughout the video. CoMO4D combines monocular depth estimates with semantic cues to construct object-level point-cloud sequences. At its core is a spatiotemporal completion mechanism that integrates semantic-region and inter-object relational reasoning, cross-frame temporal reasoning, and adaptive gated fusion to aggregate complementary observations for canonical geometry reconstruction and consistent frame-wise object-state prediction. We further construct CoScene4D, a CARLA-based 4D dataset for structured semantic, geometric, spatial, and temporal supervision. CoMO4D is trained using CARLA-based CoScene4D together with nuScenes, while V2X-Sep, Waymo, and PandaSet form three held-out test domains. Quantitative pose and foreground-appearance results on all three test datasets support cross-domain transfer of the reconstructed 4D object representation.

1 Introduction

4D content creation broadly follows two paradigms: generation and reconstruction. Generative methods synthesize dynamic 3D content from sparse prompts by relying mainly on learned priors, while reconstruction methods recover a specific scene from sufficiently informative observations. Our setting lies between the two. Given only a monocular video, observations are incomplete under occlusion and limited viewpoints. We therefore formulate the task as observation-conditioned generative 4D reconstruction: visible evidence anchors the result to the observed scene, while learned priors complete unobserved geometry and object states.

Multi-object 4D reconstruction requires persistent semantic identities, object geometry, and frame-wise spatial states. Scene-centric methods instead organize outputs around holistic geometry, points, trajectories, or motion fields, leaving the canonical geometry and states of individual objects less accessible.

Object-centric methods move closer to this goal, but occlusion and viewpoint changes leave monocular object geometry partial and independently estimated states ambiguous. Two complementary cues remain underused: within-frame relations among co-visible objects and their supporting environment, and identity-aligned observations of the same object across frames.

Figure 1 makes these two sources of evidence concrete: panel (a) shows how successive observations reveal complementary parts of moving vehicles, while panel (b) shows how spatial relations and temporal continuity constrain their evolving states.

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.

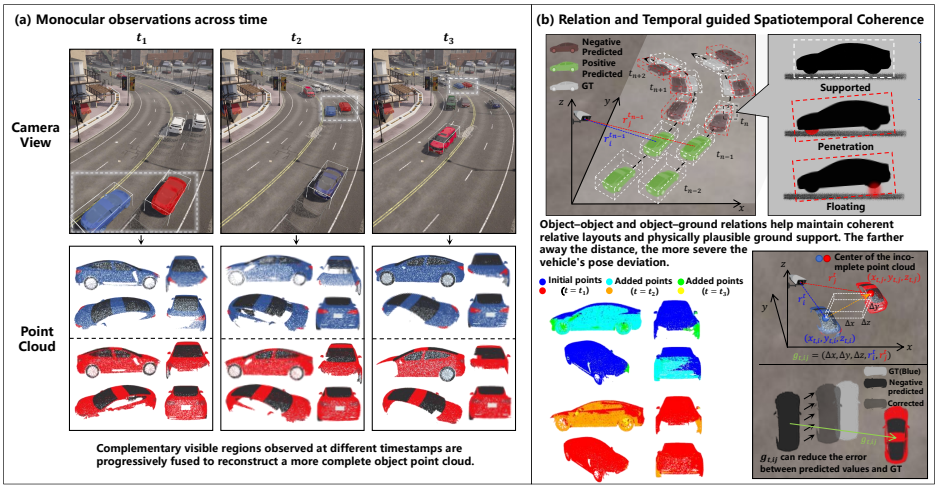

Figure 1: Motivation for complementary temporal and relational reasoning in CoMO4D. (a) Different frames reveal complementary visible surfaces of moving vehicles, enabling more complete object geometry. (b) Object–object and object–scene relations constrain relative pose and contact, while temporal continuity reduces abrupt pose drift.

Our key insight is that these two sources of evidence address different parts of the ambiguity and should be modeled explicitly but in parallel. Within each frame, semantically meaningful local regions and neighboring objects provide spatial references for object-state estimation. Across frames, the same persistent object provides complementary observations that help recover missing geometry and stabilize its evolving state. Based on this insight, we propose **CoMO4D**, which represents each persistent object with one shared canonical geometry and a sequence of frame-wise translation, orientation, and scale states. A Relation Module couples each dynamic object with nearby semantic-region context and co-visible objects, while a Consistent Object Temporal Module performs current-to-other-frame aggregation over identity-aligned observations. Their outputs are combined with the original object representation through adaptive gated fusion.

To support this setting, we further construct **CoScene4D**, a CARLA-based 4D dataset with structured semantic, geometric, spatial, and temporal supervision. CoMO4D is trained on CARLA-based CoScene4D together with nuScenes, while V2X-Sep, Waymo, and PandaSet are reserved as three external test domains. Pose and foreground-appearance results are available on all three test datasets. Our main contributions are:

1. We formulate monocular multi-object 4D modeling as observation-conditioned generative 4D reconstruction and propose CoMO4D, which reconstructs tracked dynamic rigid objects with shared canonical geometry and temporally aligned frame-wise spatial states from incomplete monocular observations.

2. We introduce parallel relational and temporal reasoning. The Relation Module combines semantic local-region context with intra-frame object–object interactions, while the Temporal Module performs current-to-other-frame aggregation over identity-aligned observations. Adaptive gated fusion combines these complementary cues for geometry and object-state reconstruction.

3. We construct CoScene4D with structured semantic, geometric, spatial, and temporal supervision, and evaluate cross-domain generalization by training on CARLA-based CoScene4D and nuScenes and evaluating on V2X-Sep, Waymo, and PandaSet.

2 Related Work

We group prior 4D methods by scene-centric, object-centric, and relational–temporal modeling.

2.1 Scene-Centric 4D Reconstruction

Scene-centric methods model dynamics with fields, Gaussians, trajectories, or dense geometry. D-NeRF Pumarola et al. (2021) requires per-scene optimization, while Dynamic 3D Gaussians Luiten et al. (2024) and 4D Gaussian Splatting Wu et al. (2024) are commonly developed with calibrated multi-view observations. GFlow Wang et al. (2025b) uses monocular depth and optical flow, and Shape of Motion Wang et al. (2025a) reconstructs long-range 3D trajectories. Feed-forward Geo4D Jiang et al. (2025), MoRe Fang et al. (2026), and Any4D Karhade et al. (2026) estimate geometry and motion using pretrained backbones. These representations rarely expose semantic objects with shared canonical geometry and frame-wise states.

2.2 Object-Centric 4D Generation and Modeling

LASR Yang et al. (2021) uses per-video optimization for articulated objects, while Motion 3-to-4 Chen et al. (2026) predicts mesh trajectories; both mainly target a single object. DreamScene4D Chu et al. (2024) and Robust4DGen Chu et al. (2025) construct per-object Gaussians from tracks for multi-object generation, but primarily recompose independently modeled entities. 4D Primitive-Mâché Mazur et al. (2026) groups rigid primitives by motion without semantic object alignment. CoMO4D instead couples relations among tracked dynamic rigid objects with their identity-aligned states.

2.3 Relational and Temporal Modeling

Lift4D Litman et al. (2026) combines single-view 3D estimation, diffusion priors, and test-time optimization for temporally consistent dynamic-object completion, but does not explicitly model relations among multiple persistent semantic objects.

SlotFormer Wu et al. (2023) and Object-Centric Video Prediction via Decoupling of Object Dynamics and Interactions Villar-Corrales et al. (2023) model object dynamics and interactions in learned latent spaces. Both target future video prediction rather than metric multi-object 4D generation with canonical geometry and frame-wise spatial states. MIDI Huang et al. (2025) improves inter-object coherence in static 3D generation but does not model temporal evolution, while DSG-DETR Feng et al. (2023) predicts evolving discrete relations over 2D object tracklets rather than continuous 3D object states. CoMO4D instead uses identity-aligned 3D object–time representations to couple intra-frame relations with cross-frame state evolution.

3 Method

3.1 Object-Level Point Cloud Construction

As shown in Figure 2, CoMO4D takes a monocular RGB video $\mathcal{I} = \{I_t\}_{t=1}^{T}$ as its raw input. A preprocessing stage obtains frame-wise depth maps $\mathcal{D} = \{D_t\}$, instance masks $\mathcal{M} = \{M_{t,n}\}$, temporally consistent Track IDs, and camera intrinsics and poses from the video. These quantities are treated as processed auxiliary information rather than additional sensing inputs. The camera parameters provide geometric conditioning for viewpoint-aware feature extraction and coordinate transformation.

For each frame, valid depth pixels are back-projected into the camera coordinate system. We adopt a point-based object representation, following common practice in point-cloud learning Qi et al. (2017a;b); Zhao et al. (2021); Misra et al. (2021), and use the instance mask to select the points belonging to object n:

$$\mathcal{P}_{t,n} = \left\{ D_t(u,v) K_t^{-1} [u, v, 1]^\top \;\middle|\; M_{t,n}(u,v) = 1 \right\}. \tag{1}$$

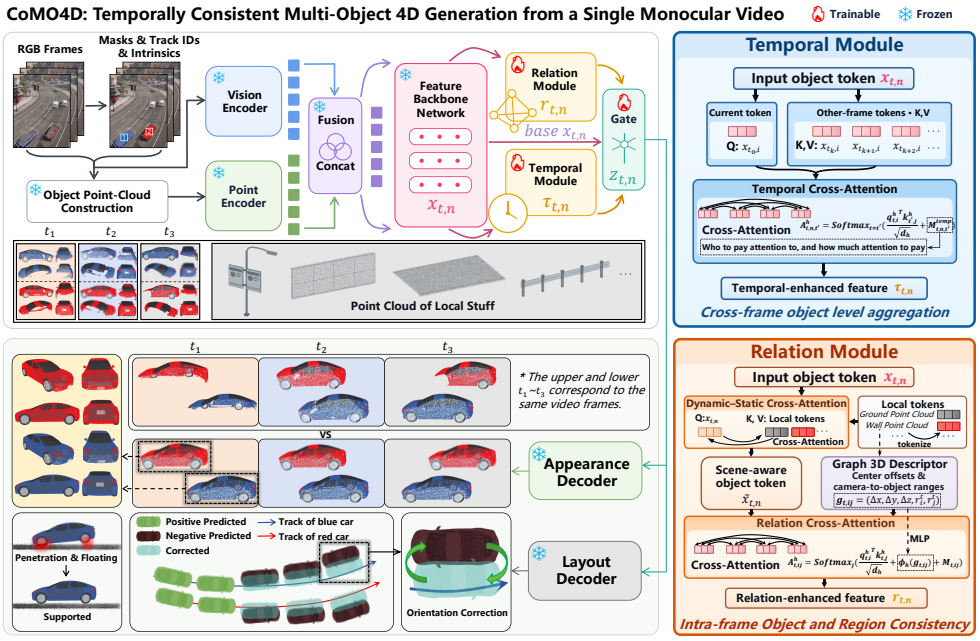

Figure 2: Overview of the CoMO4D pipeline. CoMO4D constructs object-level representations from a monocular video, performs semantic-region-conditioned relational modeling and current-to-other-frame temporal modeling with adaptive gated fusion, and generates canonical geometry together with frame-wise object translation, orientation, and scale.

The resulting object-level point cloud $\mathcal{P}_{t,n}$ represents the visible object surface in frame t. The partial observations are organized across frames to provide geometric evidence for subsequent relational, temporal, and geometry modeling.

3.2 Object-Level Point Cloud Feature Extraction

RGB features provide appearance and semantic cues, while object point-cloud features describe visible geometric structure. After projection into a shared hidden space, the visual tokens $\mathbf{V}_{t,n}$ and geometric tokens $\mathbf{G}_{t,n}$ are concatenated to form the Object Token $\mathbf{O}_{t,n}$:

$$\mathbf{O}_{t,n} = [\mathbf{V}_{t,n}; \mathbf{G}_{t,n}]. \tag{2}$$

To obtain a fixed-dimensional representation for subsequent relation and temporal modeling, we prepend a shared learnable aggregation token $\mathbf{q}^{\text{obj}}$ Dosovitskiy et al. (2021). This compact aggregation is also related to learnable query and latent representations used in DETR and Perceiver Carion et al. (2020); Jaegle et al. (2021). Frame-specific temporal and camera embeddings are added to identify the timestamp and viewpoint. The conditioned token sequence is then processed by a lightweight Transformer:

$$\mathbf{H}_{t,n} = \text{Transformer}\big([\mathbf{q}^{\text{obj}} + \mathbf{e}_t^{\text{time}} + \mathbf{e}_t^{\text{cam}} \; ; \; \mathbf{O}_{t,n} + \mathbf{e}_t^{\text{time}} + \mathbf{e}_t^{\text{cam}}]\big)\,. \tag{3}$$

Here, $\mathbf{H}_{t,n} \in \mathbb{R}^{(L+1)\times C}$ contains the updated aggregation token and L updated local tokens. We define the first output token as the fixed-dimensional object representation, $\mathbf{x}_{t,n} = \mathbf{H}_{t,n}[0] \in \mathbb{R}^C$, while $\mathbf{H}_{t,n}[1{:}]$ retains fine-grained appearance and geometric evidence for canonical geometry generation. Thus, $\mathbf{x}_{t,n}$ is a compact summary derived from the complete token sequence $\mathbf{H}_{t,n}$ rather than an independent feature.

3.3 Relational and Temporal Modeling

We collect all object representations into the object–time tensor

$$\mathbf{X} = [\mathbf{x}_{t,n}]_{t=1,n=1}^{T,N} \in \mathbb{R}^{B\times T\times N\times C}. \tag{4}$$

Although each object token contains its own appearance and geometric information, independently formed representations do not explicitly encode the surrounding semantic context, the spatial configuration of co-visible objects, or complementary observations of the same persistent object at other timestamps. CoMO4D therefore feeds the original object representations into two parallel branches. The Relation Module first relates each dynamic object to nearby semantic local regions and then to other valid objects in the same frame. In parallel, the Temporal Module uses the current object token to retrieve information from other timestamps of the same Track. Their outputs are subsequently combined with the original object representation through channel-wise adaptive gated fusion.

Region-Conditioned Relation Module.

For each frame, the Relation Module uses two spatial references: semantic local regions, such as ground, walls, and guardrails that are not compact object instances, provide contextual geometry; co-visible objects provide relative layout cues. This extends relational and scene-graph reasoning Hu et al. (2018); Zellers et al. (2018); Wald et al. (2020) to partial 3D observations.

Using processed depth, semantic labels, instance masks, and camera parameters, we select nearby region pixels $\Omega_{t,n}^{\mathrm{reg}}$ for object n and back-project them as in Eq. 1 to obtain $\mathcal{S}_{t,n}$. Semantic labels group the regions, and $\mathrm{MLP}_{\mathrm{reg}}$ maps their sampled 3D points to tokens $\mathbf{s}_{t,n,k}$ used as cross-attention keys and values.

We then perform dynamic–static cross-attention. The current object representation provides the query, while the local semantic-region tokens provide keys and values:

$$\alpha_{t,n,k} = \mathrm{Softmax}_k\left(\frac{\mathbf{q}_{t,n}^{\top}\mathbf{k}_{t,n,k}^{s}}{\sqrt{d}} + M_{t,n,k}^{s}\right), \tag{5}$$

$$\mathbf{c}_{t,n}^{s} = \sum_{k} \alpha_{t,n,k}\mathbf{v}_{t,n,k}^{s}. \tag{6}$$

The resulting semantic-region context is introduced through a residual update,

$$\widetilde{\mathbf{x}}_{t,n} = \mathrm{LN}\left(\mathbf{x}_{t,n} + \boldsymbol{\gamma}_{t,n} \odot W_s\mathbf{c}_{t,n}^{s}\right). \tag{7}$$

When the region is the supporting ground, its geometry supplies height and contact cues that help reduce penetration and floating.

After incorporating local semantic-region context, we first construct the point-cloud-based pairwise relation used for object–object reasoning. Let $\mathbf{c}_{t,i}^{c}$ and $\mathbf{c}_{t,j}^{c}$ denote the centers of the currently observed partial point clouds of objects i and j in the camera coordinate system, respectively. We define

$$\mathbf{g}_{t,ij} = [\Delta x_{t,ij}, \Delta y_{t,ij}, \Delta z_{t,ij}, r_{t,i}, r_{t,j}] \in \mathbb{R}^5, \tag{8}$$

where $[\Delta x_{t,ij}, \Delta y_{t,ij}, \Delta z_{t,ij}]^{\top} = \mathbf{c}_{t,j}^{c} - \mathbf{c}_{t,i}^{c}$, $r_{t,i} = \|\mathbf{c}_{t,i}^{c}\|_2$, and $r_{t,j} = \|\mathbf{c}_{t,j}^{c}\|_2$. Thus, $\mathbf{g}_{t,ij}$ encodes the pairwise center offset and both camera distances using only the current partial point clouds.

Conditioned on this relation vector, the Relation Module then performs object–object cross-attention. For a current object i, the region-conditioned representation $\widetilde{\mathbf{x}}_{t,i}$ acts as the query, while the region-conditioned representations of the other valid co-visible objects act as keys and values:

$$A_{t,ij}^{h} = \mathrm{Softmax}_j\left(\frac{\mathbf{q}_{t,i}^{h\,\top}\mathbf{k}_{t,j}^{h}}{\sqrt{d_h}} + \Phi_h(\mathbf{g}_{t,ij}) + M_{t,ij}\right). \tag{9}$$

The validity mask is defined by

$$M_{t,ij} = \begin{cases} 0, & \text{if } i \text{ and } j \text{ are valid at } t \text{ and } i \neq j, \\ -\infty, & \text{otherwise.} \end{cases} \tag{10}$$

The Relation Module therefore combines dynamic–static context from semantic local regions with object–object spatial relations and produces the relation-enhanced representation $\mathbf{r}_{t,n}$.

Temporal Module.

Unlike tracking-oriented cross-frame association Meinhardt et al. (2022); Zeng et al. (2022); Doersch et al. (2023); Karaev et al. (2024), the Temporal Module uses cross-frame evidence for geometry

and spatial-state recovery. It queries valid observations of the same Track at other timestamps with the current original token $\mathbf{x}_{t,n}$, rather than its region-conditioned update:

$$A^h_{t,n,t'} = \mathrm{Softmax}_{t'}\left(\frac{\mathbf{q}^h_{t,n}{}^\top \mathbf{k}^h_{t',n}}{\sqrt{d_h}} + M^{\mathrm{temp}}_{t,n,t'}\right). \tag{11}$$

The temporal validity mask is defined as

$$M^{\mathrm{temp}}_{t,n,t'} = \begin{cases} 0, & \text{if } n \text{ is valid at } t \text{ and } t', \text{ and } t' \neq t, \\ -\infty, & \text{otherwise.} \end{cases} \tag{12}$$

The mask excludes missing and current-frame slots, allowing valid same-Track observations to provide complementary evidence for the temporally enhanced representation $\boldsymbol{\tau}_{t,n}$.

Adaptive Gated Fusion.

Inspired by adaptive branch selection Li et al. (2019) and gated fusion Arevalo et al. (2017), the gate predicts weights from the concatenated base, relation, and temporal representations and normalizes them across branches for each feature channel. The fused representation is

$$\mathbf{z}_{t,n} = \mathrm{LN}\big(\boldsymbol{\alpha}^b_{t,n} \odot \mathbf{x}_{t,n} + \boldsymbol{\alpha}^r_{t,n} \odot \mathbf{r}_{t,n} + \boldsymbol{\alpha}^\tau_{t,n} \odot \boldsymbol{\tau}_{t,n}\big). \tag{13}$$

3.4 Pose, Scale, and Canonical Geometry Generation

The fused representation $\mathbf{z}_{t,n}$ is used for frame-wise object-state prediction and Track-level canonical geometry generation. The layout decoder predicts the object state directly from the fused representation:

$$\left[\widehat{\mathbf{u}}_{t,n}, \widehat{\boldsymbol{\rho}}_{t,n}, \widehat{\boldsymbol{\ell}}_{t,n}, \widehat{\kappa}_{t,n}\right] = f_{\mathrm{layout}}\left(\mathbf{z}_{t,n}\right). \tag{14}$$

The frozen layout decoder predicts 13 values: 3D translation, continuous 6D rotation Zhou et al. (2019), 3D log-scale, and a scalar translation-scale factor. In the camera coordinates used for training and evaluation, translation is $\widehat{\mathbf{p}}^c_{t,n} = \widehat{\kappa}_{t,n}\widehat{\mathbf{u}}_{t,n}$. The Relation and Temporal Modules and adaptive gate thus improve pose and scale by refining the conditioning feature $\mathbf{z}_{t,n}$, without updating the decoder.

For each Track, learnable Track Queries cross-attend to updated local observation tokens, fused object features, and viewpoint conditions across valid frames. The frozen geometry decoder uses this evidence to generate one shared canonical geometry; frame-wise translation, rotation, and scale place it throughout the video.

3.5 Learning Objective

Following the trainable/frozen configuration in Figure 2, the visual encoder, point encoder, feature backbone, geometry decoder, and layout decoder are kept frozen. Only the Relation Module, Temporal Module, and adaptive gating mechanism are optimized. Gradients from the frame-wise object-state losses pass through the frozen layout decoder to train these newly introduced representation modules. The complete objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{trans}} + \lambda_{\mathrm{rot}}\mathcal{L}_{\mathrm{rot}} + \lambda_{\mathrm{scale}}\mathcal{L}_{\mathrm{scale}}. \tag{15}$$

Here, $\mathcal{L}_{\mathrm{trans}}$ measures the Euclidean translation error in camera coordinates, $\mathcal{L}_{\mathrm{rot}}$ is the rotation geodesic loss, and $\mathcal{L}_{\mathrm{scale}}$ supervises the predicted three-dimensional log-scale. No separate relation, temporal-consistency, or canonical-geometry loss is introduced.

4 Experiments

We assess cross-domain quantitative performance, qualitative viewpoint transfer, the contributions of the Relation and Temporal Modules, and computational efficiency.

Table 1: Pose-related results on V2X-Sep, Waymo, and PandaSet. Best values within each dataset are shown in bold.

Method	Trans. (m) ↓	Rot. (°) ↓	Scale ↓	RelTrans (m) ↓	Vel. (m/s) ↓
V2X-Sep					
SAM 3D Forloop Team (2025)	5.7336	16.0912	0.8553	1.7793	2.3861
Any4D Karhade et al. (2026)	3.7986	11.5174	0.3885	2.8762	3.9414
MoRe Fang et al. (2026)	3.1446	13.6787	0.4404	2.5183	2.5107
CUPID	2.2004	13.7369	0.3134	2.1394	4.0184
CoMO4D (Ours)	**1.8527**	**8.9879**	**0.1062**	**1.7130**	**2.0909**
Waymo					
SAM 3D Forloop Team (2025)	3.2325	49.2011	0.4412	4.6529	2.4800
Any4D Karhade et al. (2026)	4.3986	22.4780	0.8460	7.0610	3.5136
MoRe Fang et al. (2026)	4.6866	23.4918	0.6340	7.2910	3.4627
CUPID	4.6497	21.8920	0.9747	7.5846	3.9081
CoMO4D (Ours)	**2.7504**	**6.3724**	**0.1268**	**4.40**	**2.35**
PandaSet					
SAM 3D Forloop Team (2025)	5.0302	11.2470	0.4058	4.3754	3.9705
Any4D Karhade et al. (2026)	6.9745	27.3912	0.5083	3.9878	**3.4431**
MoRe Fang et al. (2026)	5.4311	22.7567	0.8265	6.1366	5.1592
CUPID	4.9294	10.5988	0.6192	4.5924	3.5459
CoMO4D (Ours)	**2.5095**	**10.0689**	**0.1958**	**3.8413**	4.4343

4.1 Experimental Setup

CoMO4D is trained on CARLA-based CoScene4D and nuScenes and tested on V2X-Sep, Waymo, and PandaSet. Each monocular RGB video is accompanied by standardized depth, instance masks, object associations, and camera parameters. We report five pose metrics (Trans, Rot, Scale, RelTrans, and Vel) and three foreground-appearance metrics (FG-PSNR, FG-SSIM, and FG-LPIPS) on all three test datasets. Dataset inputs, evaluation protocols, scale alignment, and training settings are detailed in Appendix A.

4.2 Main Results

Tables 1 and 2 report pose and foreground-appearance results, respectively. Across all three datasets, CoMO4D achieves the lowest Trans, Rot, Scale, and RelTrans errors. Its Vel error is lowest on V2X-Sep and Waymo, while Any4D leads on PandaSet. For appearance, CoMO4D achieves the best FG-PSNR and FG-LPIPS on V2X-Sep and all three metrics on Waymo and PandaSet; CUPID has the highest V2X-Sep FG-SSIM. Together, the pose and appearance results support cross-domain transfer of structured object states and visual quality.

Beyond the available quantitative results in Tables 1 and 2, Figure 3 provides a direct visual comparison between CoMO4D and representative baselines under the same V2X-Sep scene and observation setting. The comparison focuses on the generated multi-object 4D result, including object geometry, object state, and the relative spatial configuration among objects. This visualization complements the numerical evaluation by showing how the differences between methods appear in the final structured scene output.

4.3 Qualitative Results and Ablation Studies

Multi-Scene Qualitative Results. Figure 4 shows how CoMO4D maintains coherent persistent objects across scenes with varied layouts, motions, and visibility conditions.

Table 2: Foreground-appearance results on V2X-Sep, Waymo, and PandaSet. Best values within each dataset are shown in bold.

Dataset	Method	FG-PSNR ↑	FG-SSIM ↑	FG-LPIPS ↓
V2X-Sep	SAM 3D Forloop Team (2025)	13.35	0.161	0.477
	CUPID	13.58	**0.170**	0.459
	Lift4D Litman et al. (2026)	13.25	0.158	0.508
	CoMO4D (Ours)	**13.92**	0.160	**0.323**
Waymo	SAM 3D Forloop Team (2025)	25.4344	0.5528	0.1482
	CUPID	25.1723	0.5520	0.1351
	Lift4D Litman et al. (2026)	25.4186	0.5501	0.1482
	CoMO4D (Ours)	**26.0837**	**0.5826**	**0.1193**
PandaSet	SAM 3D Forloop Team (2025)	19.1526	0.6475	0.5215
	CUPID	19.1610	0.6487	0.4978
	Lift4D Litman et al. (2026)	19.1603	0.6488	0.5262
	CoMO4D (Ours)	**20.0643**	**0.6817**	**0.4278**

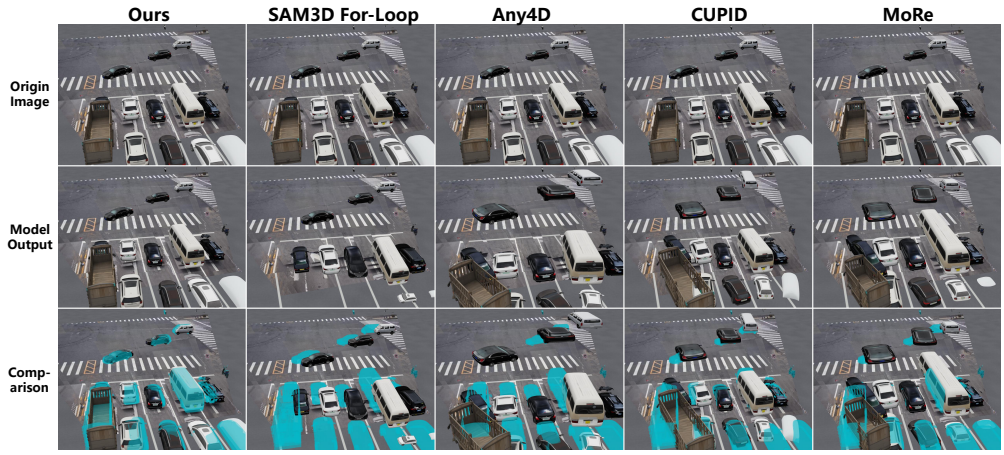

Figure 3: Qualitative comparison on the same V2X-Sep scene. CoMO4D is compared with representative baseline methods under the same observation setting, enabling direct comparison of object geometry, object states, and multi-object spatial configuration.

Object-Level Appearance Comparison. Figure 5 isolates the same object across methods to compare completeness, local shape detail, and appearance coherence.

Ablation Studies. We ablate the Relation and Temporal Modules individually and jointly under identical training settings (Table 3).

Table 3: Ablation of the Relation and Temporal Modules. Lower is better for all metrics.

Variant	Trans ↓	Rot ↓	Scale ↓	RelTrans ↓	Vel ↓
w/o Relation	1.62	11.20	0.155	1.95	0.29
w/o Temporal	1.68	11.80	0.158	1.90	0.33
w/o Both	1.82	12.80	0.170	2.05	0.36
CoMO4D (Ours)	**1.51**	**10.25**	**0.120**	**1.82**	**0.27**

CoMO4D leads on all five metrics, while removing both branches yields the largest errors. Removing Temporal degrades Trans, Rot, Scale, and Vel more than removing Relation; removing Relation

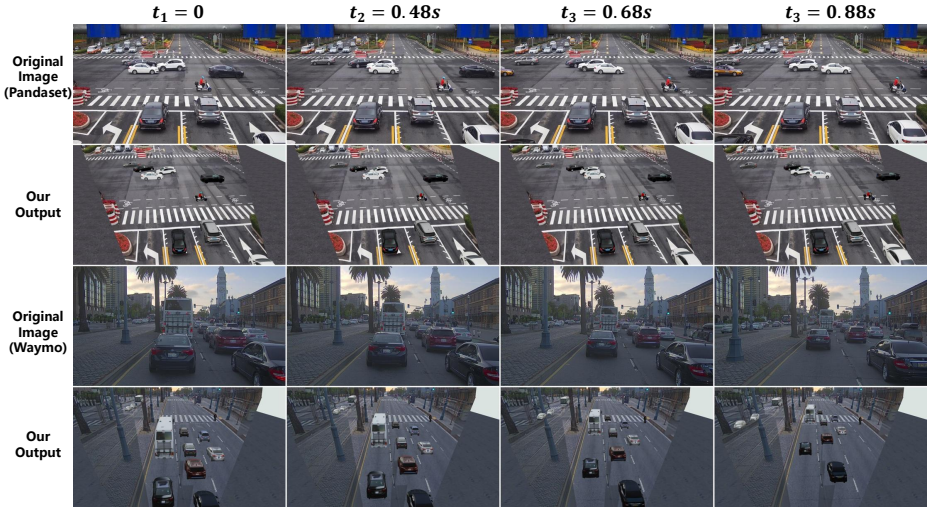

Figure 4: Qualitative results on PandaSet (top) and Waymo (bottom). The examples show object reconstruction across consecutive frames in two real-world test datasets.



Figure 5: Object-level appearance comparison on V2X-Sep. The same object is compared across CoMO4D and representative baseline methods to visualize differences in object completeness, local geometric detail, and appearance coherence.

increases RelTrans more (1.95 versus 1.90). These differences match the branches' cross-frame and intra-frame roles, respectively.

CoMO4D also reduces average mesh-generation time by approximately 6.7× relative to SAM 3D Forloop on the same 30-clip protocol without increasing peak GPU memory (Appendix A.5).

5 Conclusion

CoMO4D performs observation-conditioned generative 4D reconstruction from monocular video, combining identity-aligned temporal observations with object and semantic-region relations to recover canonical geometry and frame-wise states under occlusion. We also introduce CoScene4D, a CARLA-based dataset and benchmark to be released with our implementation. Pose and foreground-appearance results on V2X-Sep, Waymo, and PandaSet support cross-domain transfer.

REFERENCES

John Arevalo, Thamar Solorio, Manuel Montes-y Gómez, and Fabio A. González. Gated multimodal units for information fusion. In *International Conference on Learning Representations Workshops*, 2017.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, pages 213–229, 2020. doi: 10.1007/978-3-030-58452-8_13.

Hongyuan Chen, Xingyu Chen, Zexiang Xu, and Anpei Chen. Motion 3-to-4: 3d motion reconstruction for 4d synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28947–28958, 2026.

Wen-Hsuan Chu, Lei Ke, and Katerina Fragkiadaki. DreamScene4D: Dynamic multi-object scene generation from monocular videos. In *Advances in Neural Information Processing Systems*, volume 37, pages 96181–96206, 2024.

Wen-Hsuan Chu, Lei Ke, Jianmeng Liu, Mingxiao Huo, Pavel Tokmakov, and Katerina Fragkiadaki. Robust multi-object 4d generation for in-the-wild videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22067–22077, 2025.

Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. TAPIR: Tracking any point with per-frame initialization and temporal refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10061–10072, 2023.

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. In *International Conference on Learning Representations*, 2021.

Juntong Fang, Zequn Chen, Weiqi Zhang, Donglin Di, Xuancheng Zhang, Chengmin Yang, and Yu-Shen Liu. MoRe: Motion-aware feed-forward 4d reconstruction transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28914–28924, 2026.

Shengyu Feng, Hesham Mostafa, Marcel Nassar, Somdeb Majumdar, and Subarna Tripathi. Exploiting long-term dependencies for generating dynamic scene graphs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5130–5139, 2023.

Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018.

Zehuan Huang, Yuan-Chen Guo, Xingqiao An, Yunhan Yang, Yangguang Li, Zi-Xin Zou, Ding Liang, Xihui Liu, Yan-Pei Cao, and Lu Sheng. MIDI: Multi-instance diffusion for single image to 3d scene generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23646–23657, 2025.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4651–4664, 2021.

Zeren Jiang, Chuanxia Zheng, Iro Laina, Diane Larlus, and Andrea Vedaldi. Geo4D: Leveraging video generators for geometric 4d scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20658–20671, 2025.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker: It is better to track together. In *Computer Vision – ECCV 2024*, 2024.

Jay Karhade, Nikhil Keetha, Yuchen Zhang, Tanisha Gupta, Akash Sharma, Sebastian Scherer, and Deva Ramanan. Any4D: Unified feed-forward metric 4d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14578–14589, 2026.

Marius Kästingschäfer, Théo Gieruc, Sebastian Bernhard, Dylan Campbell, Eldar Insafutdinov, Eyvaz Najafli, and Thomas Brox. Seed4d: A synthetic ego–exo dynamic 4d data generator, driving dataset and benchmark. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 7741–7753, 2025.

Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 510–519, 2019.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3758–3768, 2023.

Yehonathan Litman, Xiaoxuan Ma, Manan Shah, Nicolas Ugrinovic, Kris Kitani, Fernando De la Torre, and Shubham Tulsiani. Lift4D: Harmonizing single-view 3d estimation for 4d reconstruction in-the-wild, 2026.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21013–21022, 2022.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *Proceedings of the International Conference on 3D Vision*, pages 800–809, 2024.

Kirill Mazur, Marwan Taher, and Andrew J. Davison. 4d primitive-mâché: Glueing primitives for persistent 4d scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7372–7381, 2026.

Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixé, and Christoph Feichtenhofer. TrackFormer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8844–8854, 2022.

Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2906–2917, 2021.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017a.

Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, volume 30, 2017b.

SAM-3D Team. Sam-3d: Segment anything in 3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18901–18912, 2025.

Hao-Tang Tsui, Yu-Rou Tuan, Ethan Lai, and Chen-Yu Wang. Stady4d: Towards complete 4d static–dynamic reconstruction with sigma. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 4697–4707, 2026.

Basile Van Hoorick, Purva Tendulkar, Dídac Surís, Dennis Park, Simon Stent, and Carl Vondrick. Revealing occlusions with 4d neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3011–3021, 2022.

Angel Villar-Corrales, Ismail Wahdan, and Sven Behnke. Object-centric video prediction via decoupling of object dynamics and interactions. In *Proceedings of the IEEE International Conference on Image Processing*, pages 570–574, 2023. doi: 10.1109/ICIP49359.2023.10222810.

Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3961–3970, 2020.

Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9660–9672, 2025a.

Shizun Wang, Xingyi Yang, Qiuhong Shen, Zhenxiang Jiang, and Xinchao Wang. GFlow: Recovering 4d world from monocular video. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7862–7870, 2025b. doi: 10.1609/aaai.v39i8.32847.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20310–20320, 2024.

Ziyi Wu, Nikita Dvornik, Klaus Greff, Thomas Kipf, and Animesh Garg. SlotFormer: Unsupervised visual dynamics simulation with object-centric models. In *International Conference on Learning Representations*, 2023.

Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Huiwen Chang, Deva Ramanan, William T. Freeman, and Ce Liu. LASR: Learning Articulated Shape Reconstruction from a Monocular Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15975–15984, 2021.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2018.

Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. MOTR: End-to-end multiple-object tracking with transformer. In *Computer Vision – ECCV 2022*, 2022.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16259–16268, 2021.

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*, pages 5745–5753, 2019.

A Experimental Protocol and Implementation Details

A.1 Datasets and Inputs

We use CARLA-based CoScene4D together with nuScenes for model training. CoScene4D is described in Appendix B; its structured object-level annotations provide semantic, geometric, spatial, and temporal supervision. Neither CARLA nor nuScenes is used as a quantitative test domain. Instead, V2X-Sep, Waymo, and PandaSet form three external test datasets for transfer evaluation. Each pose sample is an eight-frame monocular clip. V2X-Sep contains 750 test clips from 21 sequences at five roadside intersections, sampled at 10 fps. PandaSet contains 774 test clips from 41 driving scenes, also at 10 fps. The Waymo vehicle evaluation uses clips from its three forward-facing cameras and retains annotated vehicles at 1–50 m. CoMO4D uses depth estimated from RGB by a monocular model, together with prepared object associations, instance masks, and camera parameters. LiDAR scans, projected LiDAR depth, and 3D boxes are not passed to CoMO4D. Dataset-specific sensor annotations are reserved for reference construction and evaluation: V2X-Sep projects roadside LiDAR for reference depth and ground planes, PandaSet uses derived masks for object matching, and Waymo projects LiDAR boxes to create evaluation mask proxies. Baselines receive the prepared inputs required by their respective protocols.

A.2 Evaluation Settings

The three test datasets use the same five pose-metric columns: Translation Error (Trans, m), Rotation Error (Rot, degrees), Scale Error (Scale, unitless), Relative Translation Error (RelTrans, m), and Velocity Error (Vel, m/s). The appearance table uses FG-PSNR, FG-SSIM, and FG-LPIPS. Pose and appearance measurements are available for all three test datasets. Lower values are better for Trans, Rot, Scale, RelTrans, Vel, and FG-LPIPS, whereas higher values are better for FG-PSNR and FG-SSIM. The five pose metrics and their evaluation targets are summarized in Table 7. Pose comparisons use object–frame slots with valid ground truth and valid predictions from all five pose methods, so each method is evaluated on the same slots. On V2X-Sep, the five-way intersection contains 17,621 valid vehicle–frame observations across 750 clips: 15,701 cars, 814 vans, 569 trucks, and 537 buses. It contains 2,223 clip-level object tracks eligible for Vel. On PandaSet, the comparison uses forward-facing objects within 35 m and excludes pedestrians, leaving 2,038 common valid object–frame observations across 774 clips (2,793 before removing 755 pedestrian observations). These yield 264 eligible tracks for Vel and 2,961 within-frame object pairs for RelTrans. Waymo evaluates vehicles within 1–50 m and has 378 common valid vehicle–frame observations. These object–frame and clip-level track counts are distinct from the number of unique physical objects.

A.3 Baseline Protocols and Scale Alignment

We compare CoMO4D with SAM 3D Forloop Team (2025), Any4D Karhade et al. (2026), MoRe Fang et al. (2026), CUPID, and Lift4D Litman et al. (2026). The pose and appearance tables report methods with results for their respective metric families. The pose-evaluation pipeline converts method-native outputs to a common camera convention, performs dataset-specific scale alignment and validation-only yaw calibration, and then computes metrics on common valid object–frame slots. Scale factors apply jointly to translation and object dimensions, never to rotation. CoMO4D retains its native metric scale; the other methods follow their respective ground-anchored, internally scaled, or native-scale protocols described below.

The SAM 3D Forloop entry follows a frame-wise for-loop protocol: for every valid frame–object slot, the official SAM 3D Objects stage-1 model independently processes the RGB frame and the shared instance mask, using its own RGB-derived point map. Its predicted camera-frame pose is converted to our common camera convention, and its canonical point-cloud extent is scaled to obtain the three-dimensional object size. No state is passed between frames; Track IDs index the shared object slots but are not fed to the SAM 3D Objects model. A category-wise yaw offset is calibrated on held-out validation data for evaluation.

Common coordinate and rotation convention. Before evaluation, native predictions and reference states are expressed in a common camera frame with x forward, y left, and z up. Outputs stored

with y pointing right are converted by

$$S = \mathrm{diag}(1, -1, 1), \qquad \mathbf{t}^{\mathrm{ccf}} = S\mathbf{t}^{\mathrm{src}}, \qquad R^{\mathrm{ccf}} = SR^{\mathrm{src}}S. \tag{16}$$

The six-dimensional rotation representation is reconstructed from the first two columns of the converted rotation matrix, rather than flipping its coordinates independently. Method outputs already stored in this common frame are not converted again. For vehicle heading, a 180° reversal is treated as equivalent. Only SAM 3D Forloop and CUPID receive a category-wise constant yaw offset estimated on held-out validation data; the vehicle offset uses a doubled-angle circular mean. CoMO4D uses the identity offset, while Any4D and MoRe receive no yaw calibration. If no held-out validation set is available, the offset is zero; no offset is fitted on test annotations.

V2X-Sep ground anchoring. The V2X-Sep cache supplies a ground plane $\mathbf{n}^\top \mathbf{p} + d = 0$ derived from roadside LiDAR. For each of the four predicted box-bottom corners $\mathbf{B}_k$ (the face whose local normal is most opposite gravity), ground contact gives the candidate $\alpha_k = -d/(\mathbf{n}^\top \mathbf{B}_k)$. A single scale factor for each method and clip is the weighted median of valid positive candidates, with weight $|\mathbf{n}^\top \mathbf{B}_k|$, without residual-based rejection; the same factor applies to all its object translations and dimensions, but not rotations. CoMO4D retains its native metric scale. SAM 3D Forloop and CUPID use the ground-anchored factor, while MoRe's corresponding factor is already applied by its adapter and is not applied twice. Any4D uses a fixed ground-scale factor of 2.1672 obtained from ground-plane probes. No factor is fitted to ground-truth 3D boxes.

PandaSet mask anchoring. PandaSet constructs a ground plane for each frame from the calibrated camera pose, assuming road height near zero in world coordinates. Its pose comparison is limited to forward depth at most 35 m after excluding pedestrians; at greater distances, depth-derived object geometry is less reliable under this protocol. To avoid selecting a predicted box bottom from an uncertain orientation, the contact anchor is taken from the instance mask: its bottom row is the 97.5th percentile of mask-pixel row coordinates, and the nearby column is their median. The corresponding camera ray intersects the ground plane at $\mathbf{P}$. With predicted raw translation $\mathbf{t}_{\mathrm{raw}}$, ground normal $\mathbf{n}$, and h equal to the smallest predicted object dimension, the candidate scale is

$$\alpha_{\mathrm{mask}} = \frac{\mathbf{P}^\top \widehat{\mathbf{u}}}{\|\mathbf{t}_{\mathrm{raw}}\|_2 - \frac{h}{2}\mathbf{n}^\top \widehat{\mathbf{u}}}, \qquad \widehat{\mathbf{u}} = \frac{\mathbf{t}_{\mathrm{raw}}}{\|\mathbf{t}_{\mathrm{raw}}\|_2}. \tag{17}$$

The clip-level factor is the unweighted median over valid object–frame candidates. CoMO4D is kept at native metric scale; SAM 3D Forloop, MoRe, and CUPID use the applicable mask-ground factor. Any4D is left unscaled when its dataset-level factor lies within 0.2 of one. PandaSet reference orientation is reconstructed with the annotated length axis as heading, width as the second axis, and up as the third, rather than using the cached axis-permuted rotation field.

Waymo mask anchoring. Waymo uses the same mask-bottom construction and Eq. 17, with the ground plane supplied by its calibrated vehicle cache. Ground-truth translation is the geometric center of the LiDAR box, so neither the reference nor native prediction is shifted upward from a ground pivot. Native CoMO4D and SAM 3D Forloop pose files receive the axis conversion in Eq. 16; frozen Any4D, MoRe, and CUPID common-frame outputs are not flipped again. Any scale previously embedded in an adapter output is first removed, then the common mask-ground factor is recomputed for each method and clip. This factor is applied to SAM 3D Forloop, MoRe, and CUPID. CoMO4D remains at native metric scale; Any4D uses a dataset-level factor only when its median mask-ground estimate differs from one by more than 0.2. In every case, a selected scale multiplies both translation and object dimensions, never rotation.

A.4 Training Settings

We train CoMO4D using the AdamW optimizer with an initial learning rate of 1×10^{-4}. The batch size is set to 8, and the model is trained for 5 epochs using cosine annealing with a 5% linear warm-up. Each training sample contains an eight-frame monocular clip. The default loss weights are $\lambda_{\mathrm{rot}} = 0.2$ and $\lambda_{\mathrm{scale}} = 0.1$. Model checkpoints and hyperparameters are selected on the validation split according to Translation Error. All experiments are conducted on 8 NVIDIA A100 GPUs, and the reported values are averaged over 3 independent runs.

A.5 EFFICIENCY

We evaluate whether the additional relational–temporal reasoning introduces a prohibitive computational cost. We compare CoMO4D with SAM 3D Forloop under the same hardware and inference pipeline over 30 clips, reporting average mesh-generation time, peak GPU memory utilization, and GPU-memory-seconds in Table 4.

Table 4: Efficiency comparison with SAM 3D Forloop averaged over 30 clips. Lower is better for generation time and GPU-memory-seconds.

Method	Mesh generation time ↓	Peak GPU memory	GPU-memory-seconds ↓
SAM 3D Forloop	10m 52s	26.7% (21.4 / 80 GiB)	13,944 GiB·s
CoMO4D (Ours)	**1m 38s**	26.7% (21.4 / 80 GiB)	**2,091 GiB·s**

The efficiency result is favorable rather than merely comparable. CoMO4D reduces average mesh-generation time from 10 minutes 52 seconds to 1 minute 38 seconds, approximately a $6.7\times$ reduction, while peak GPU memory remains unchanged at 26.7% (21.4 GiB on an 80-GiB GPU). Consequently, GPU-memory-seconds decrease from 13,944 to 2,091 GiB·s, again by about $6.7\times$. Thus, the gains reported above do not require higher peak GPU memory and are accompanied by substantially lower end-to-end generation cost.

B CoScene4D Dataset

We construct **CoScene4D**, a CARLA-based 4D dataset that provides structured semantic, geometric, spatial, and temporal supervision for dynamic-scene research. Although the experiments in this paper focus on temporally consistent multi-object 4D generation, the full dataset is designed as a broader resource and additionally retains static infrastructure, dense metric geometry, surface information, persistent identities, and structured spatial and functional relations.

Under the primary CoMO4D protocol, one monocular RGB stream is treated as the raw visual input. Estimated depth, instance masks, persistent Track IDs, and camera intrinsics and poses are standardized preprocessing outputs. Ground-truth depth, simulator-native geometry, and additional synchronized observations are retained for supervision, reference construction, and the broader tasks supported by CoScene4D rather than being exposed as multi-view inputs to CoMO4D.

Table 5 compares CoScene4D with CARLA-4D Van Hoorick et al. (2022), RealGraph Lin et al. (2023), HOI4D Liu et al. (2022), SEED4D Kästingschäfer et al. (2025), and StaDy4D Tsui et al. (2026) in terms of the object-centric 4D supervision explicitly provided by their official releases. A check mark denotes directly available information; indirectly derivable information is not counted. Instance-level annotation includes instance masks, boxes, or instance-associated labels. Object-level geometry requires an instance-specific mesh or dense surface reference beyond a 3D bounding box, while inter-object relations denote explicit relational annotations or independently defined evaluation targets.

B.1 SUPPORTED 4D TASKS

We introduce CoScene4D, a comprehensive synthetic benchmark for temporally consistent, object-centric 4D understanding in dynamic urban environments. Existing 4D datasets typically focus on individual aspects of scene understanding—such as dense geometry, persistent object tracking, or inter-object relationships—but rarely provide these forms of supervision within a unified framework while maintaining metric accuracy in complex, dynamic settings. To address this limitation, CoScene4D leverages the high-fidelity CARLA simulator to generate large-scale urban scenarios containing both static infrastructure and dynamic agents. The benchmark provides spatially and temporally aligned, sensor-realistic annotations for every visible object, including metric depth maps, surface normals, instance-level segmentation masks, watertight meshes, accurate 6-DoF poses, and temporally persistent object identifiers. By integrating dense object geometry, persistent identities,

Table 5: Comparison of object-centric 4D supervision

Property	CARLA-4D	RealGraph	HOI4D	SEED4D	StaDy4D	CoScene4D
Source	Synthetic driving	Real multi-view	Real RGB-D interaction	Synthetic driving	Synthetic driving	Synthetic driving
Dense metric depth	✓	×	✓	✓	✓	✓
Instance annotations	×	✓	✓	✓	✓	✓
Persistent object IDs	×	✓	✓	×	✓	✓
Object-level geometry	×	×	✓	×	×	✓
Inter-object relations	×	✓	×	×	×	✓
Per-object motion	×	✓	✓	×	✓	✓

metric spatial states, inter-object relationships, and object-level motion into a single controllable and photorealistic environment, CoScene4D establishes a unified testbed for comprehensive 4D scene understanding.

With its unified ground-truth annotations spanning geometry, semantics, identity, relationships, and motion, CoScene4D enables comprehensive supervision and evaluation across multiple dimensions of dynamic scene perception. The current benchmark supports four core tasks:

4D Open-Vocabulary Scene Understanding. This task leverages instance segmentation, semantic labels, and persistent IDs to evaluate open-vocabulary recognition, segmentation, and temporal association, with an emphasis on semantic generalization and identity consistency.

Dynamic Object-Level Geometric Reconstruction. Using watertight meshes, metric depth, and surface normals, this task evaluates the reconstruction of complete object-level 3D geometry from videos under object motion, camera movement, and occlusion.

Temporal 6-DoF Pose Estimation and Tracking. Based on frame-wise poses, persistent IDs, and motion trajectories, this task assesses continuous 6-DoF object localization and tracking in terms of pose accuracy, temporal consistency, and identity preservation.

Structured Scene Graph Generation. Leveraging holistic scene annotations and inter-object relationships, this task evaluates the inference of spatial and functional relations among entities to construct interpretable scene graphs for high-level reasoning and planning.

B.2 Data Generation and Annotation

CoScene4D is generated from CARLA dynamic environments containing vehicles, pedestrians, and cyclists. Scenes are sampled from five CARLA towns under diverse traffic densities, actor configurations, motion patterns, weather conditions, and illumination settings.

Synchronized cameras capture RGB images, metric depth maps, instance segmentation, and semantic segmentation. Each frame is also associated with camera intrinsics and poses, persistent actor identities, object poses, physical dimensions, semantic categories, and visibility information. CARLA actor IDs are retained throughout each sequence to establish temporally consistent object tracks.

For the semantic-region-conditioned relation modeling used by CoMO4D, local region information is derived from the same processed depth, semantic labels, instance masks, and camera parameters. Points belonging to semantically meaningful extended regions are back-projected into the camera coordinate system and sampled in the neighborhood of each dynamic object to construct local region tokens for dynamic–static cross-attention.

The collected observations are organized into fixed-length video clips. All clips originating from the same physical scene are assigned to the same dataset split, preventing temporal overlap and

Table 6: CoScene4D dataset statistics.

Item	CoScene4D
Dynamic scenes	50
Benchmark clips	8,850
RGB-D view-frame observations	50,400
CARLA towns	5
Weather / illumination conditions	10+
Object groups	3
Scene-unique actor tracks	$\approx$ 2,000
Frames per scene	16
Synchronized viewpoints	63
Clip length / stride	8 / 4

Table 7: Five pose-related evaluation metrics used for the three test datasets.

Dimension	Metric	Unit	Target
Absolute layout	Translation Error	m	Per-object 3D position accuracy
Pose	Vehicle Yaw Error	$^\circ$	180°-symmetric heading accuracy
Scale	Relative Scale Error	–	Object length, width, and height
Spatial relation	Relative Translation Error	m	Relative layout in the same frame
Motion	Velocity Error	m/s	Object velocity accuracy

scene-level leakage. Under the primary monocular protocol, one RGB stream is used as the raw input, while the corresponding depth, masks, Track IDs, and camera parameters are supplied as standardized preprocessing outputs. Instance-aligned depth pixels from the remaining synchronized viewpoints are back-projected into 3D space and transformed into a shared coordinate system to construct dense object-level geometric references.

B.3 Dataset Statistics

CoScene4D contains 50 dynamic scenes and 8,850 valid clips. It covers five CARLA towns, more than ten weather and illumination conditions, three object groups, and approximately 2,000 scene-unique actor tracks. The main dataset statistics are summarized in Table 6.

The scene-disjoint 80/10/10 split contains 40 training scenes, 5 validation scenes, and 5 test scenes, corresponding to 7,080, 885, and 885 clips, respectively. Dedicated subsets are additionally constructed according to occlusion severity, motion magnitude, and scene density. A cross-town protocol holds out an unseen CARLA town to evaluate generalization to new scene layouts.

B.4 CoScene4D Evaluation

CoScene4D evaluates object layout, orientation, physical scale, intra-frame spatial relations, and motion using the five pose-related metrics in Table 7, matching the metric set in the combined pose comparison. Lower values are better for all five metrics. Among them, Vel measures changes in translation error over time. Across the full evaluation, foreground-appearance metrics and qualitative comparisons provide complementary evidence about reconstructed object geometry.

Translation Error. The mean Euclidean distance between the predicted and ground-truth three-dimensional object translations in the camera coordinate system.

Vehicle Yaw Error. Predicted 6D rotations are orthogonalized before yaw is extracted. For the evaluated vehicle classes, the angular difference from the reference yaw is folded modulo 180° and reported in $[0, 90^\circ]$; this treats a front–rear reversal as equivalent but does not remove a 90° axis error.

Relative Scale Error. After aligning dimensions with gravity, the absolute relative errors of predicted length, width, and height are averaged over the three axes and valid objects, and reported as a unitless ratio.

Relative Translation Error. The mean Euclidean error between predicted and ground-truth pairwise object displacements within the same frame, excluding self-pairs.

Velocity Error. For each Track with at least four valid frames in the common intersection, observations are ordered by timestamp and split into earlier and later windows. Let $\bar{\mathbf{e}}_1$ and $\bar{\mathbf{e}}_2$ be the mean signed translation errors $\mathbf{t}^{\mathrm{pred}} - \mathbf{t}^{\mathrm{gt}}$ in the two windows, and let $\bar{t}_1, \bar{t}_2$ be their mean timestamps. The reported error is $\|\bar{\mathbf{e}}_2 - \bar{\mathbf{e}}_1\|_2/(\bar{t}_2 - \bar{t}_1)$ in meters per second; subtracting the window means removes a track-wise constant translation bias.

C Limitations and Future Work

CoMO4D currently focuses on reconstructing dynamic rigid objects. Its shared canonical geometry and rigid frame-wise state are less suitable for deformable objects such as pedestrians and moving leaves, for which fine-scale shape changes and motion are not recovered reliably.

Future work will extend the representation to deformable objects and incorporate scene components that are difficult to isolate as individual instances, including complete walls and road surfaces. Modeling these structures together with dynamic objects would move toward a more complete and coherent 4D reconstruction of the environment.