LUCIDLYRA: FEED-FORWARD GENERATIVE 3D SCENE RECONSTRUCTION AND JOINT UNDERSTANDING VIA MULTI-TASK DISTILLATION

Anonymous authors
Paper under double-blind review

https://anonlucidlyra.github.io

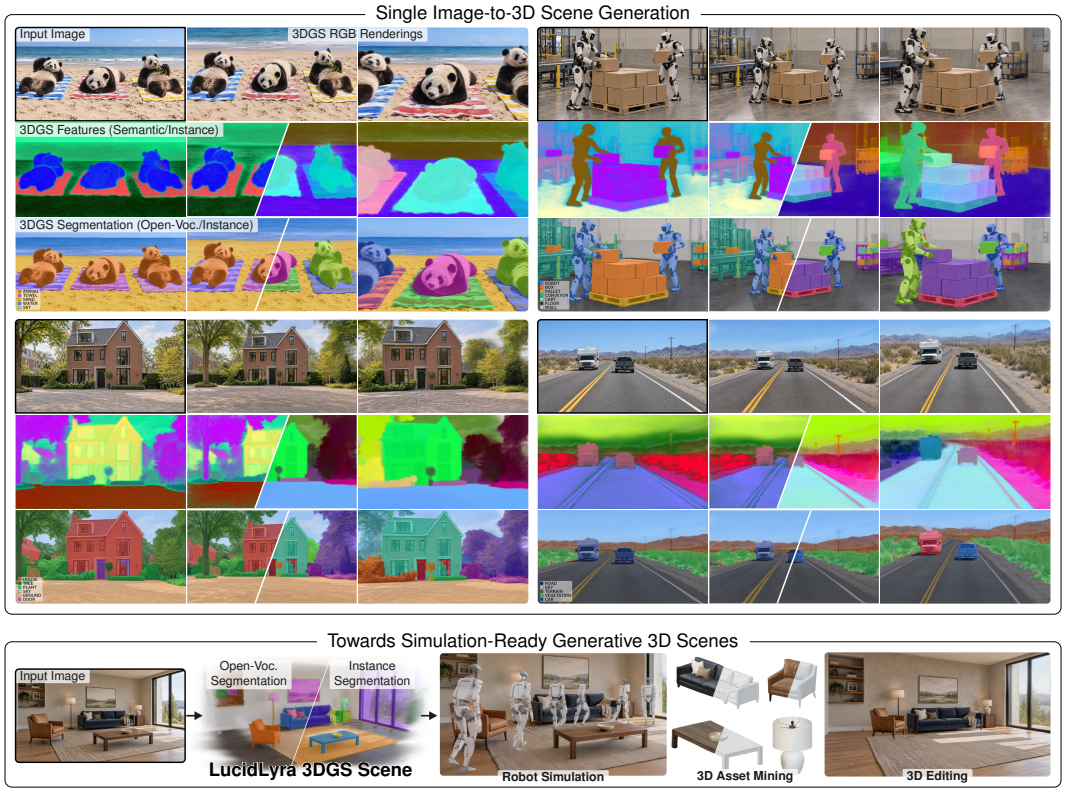


Figure 1: **Feed-forward semantics- and instance-aware image-to-3D scene generation.** *Top:* LucidLyra infers a unified 3D Gaussian Splatting (3DGS) representation consisting of appearance, geometry, semantic, and instance features from a single image. The learned feature fields enable open-vocabulary semantic segmentation and instance segmentation. *Bottom:* LucidLyra enables downstream applications such as physical-AI simulation, editing, and 3D asset mining.

ABSTRACT

We present LucidLyra, a framework that makes generatively reconstructed 3D scenes directly queryable for semantics and instances in a feed-forward manner from video latents. While generative 3D reconstruction can synthesize virtual environments from a single image, downstream applications in robotics and physical AI require not only visual realism but also access to scene semantics and object identities. Existing 3D understanding approaches, however, require captured multi-view scenes, have limited generalization capabilities, or rely on expensive per-scene processing and optimization. To bridge this gap, our method distills complementary knowledge from a camera-controlled video diffusion model and frozen 2D vision foundation models into 3D Gaussians that jointly encode geometry, appearance, open-vocabulary semantics, and class-agnostic instance

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.

cues. The resulting scenes support novel-view rendering, open-vocabulary segmentation, and object grouping without per-scene optimization, while training requires neither manual 3D annotations nor captured multi-view data. Our method outperforms baselines combining state-of-the-art 3DGS segmentation approaches with generative 3D scene reconstruction on curated synthetic and real-image-conditioned benchmarks, while preserving reconstruction quality. Furthermore, our latent formulation naturally extends to monocular video, reconstructing structured dynamic 3D scenes across time. By turning renderable but unstructured outputs into queryable 3D worlds, LucidLyra enables downstream applications including 3D scene editing, asset extraction, and robotic simulation.

1 INTRODUCTION

Recent generative reconstruction methods can turn a single image into a renderable 3D environment. Building upon learned generative models, they can imagine content beyond the observed field of view and consolidate it into a scene that can be viewed from new perspectives (Gao et al., 2024; Ren et al., 2025; Liang et al., 2025a; Szymanowicz et al., 2025; Schwarz et al., 2025; Bahmani et al., 2026). Generated 3D environments enable scalable content generation for interactive media, physical AI simulation, and industrial applications in engineering and manufacturing (He et al., 2026). However, existing generated scene representations primarily capture geometry and appearance, while lacking an explicit understanding of their content. Downstream applications in embodied AI and physical simulation cannot interact with an unsegmented collection of colored primitives; they require structured scene elements with semantic roles and discrete instance boundaries to support collision reasoning, physical manipulation, and interactive scene editing.

A natural solution might seem to generate an appearance-only scene first and recover semantic and instance representations post-hoc. However, this two-stage route breaks down in the single-view generative setting. Existing post-hoc methods either attach 2D foundation-model predictions to an existing reconstruction through per-scene optimization (Qin et al., 2024; Ye et al., 2024; Tian et al., 2025; Xiong et al., 2026), or jointly infer geometry and understanding from multiple captured views (Zhu et al., 2026; Jiang et al., 2026a; Tian et al., 2026a). The former requires expensive scene-specific foundation model preprocessing and optimization. The latter relies on captured multi-view inputs and is constrained by the limited scale and domain diversity of multi-view training data. Neither route directly yields a structured 3D scene that has been generated from a single image.

Scaling these capabilities is not merely a matter of attaching a feature head to a reconstruction model. Generated regions lack paired ground-truth supervision, so estimated geometry, semantics, and object boundaries can drift and disagree across views. Predictions from 2D vision models compound this, as their semantics and masks are not guaranteed to be view-consistent. The central challenge is to turn complementary but imperfect generative and perceptual signals into a consistent 3D representation while balancing geometry, appearance, semantics, and instances without compromising generation quality. An integrated model should let semantic priors and geometry reinforce each other, rather than tagging an isolated reconstruction with semantic or object information.

We introduce LucidLyra, a framework that unifies feed-forward generative 3D reconstruction and scene understanding in an explicit Gaussian representation (Fig. 1). Our central insight is that a video generator provides a unified foundation for structured 3D scene generation, serving both as the shared backbone and as the source of training data. Building on Lyra's latent-to-Gaussian formulation (Bahmani et al., 2026), LucidLyra learns a 3D scene representation from camera-controlled video latents through multi-task distillation. Given a single input image, we denoise the corresponding video latents across camera trajectories, and process them through two parallel branches. The teacher branch decodes the latent into an RGB video and applies frozen vision-language, segmentation, and tracking foundation models to obtain semantic and instance supervision. The student branch incorporates vision cues of the input view into the video latents and directly maps the shared latent into a 3D Gaussian scene. We consolidate the teacher predictions across the generated views and distill them into persistent semantic and instance fields on the Gaussian primitives, jointly with geometry and appearance. A single shared representation can therefore support novel-view rendering, open-vocabulary segmentation, and object grouping. As an initial exploration toward generative 4D scene understanding, we also extend the formulation to monocular video inputs and predict

semantics- and instance-aware dynamic 3D scenes. For this novel task, we introduce an evaluation protocol with benchmarks covering static scenes, real-image domain transfer, and dynamic scenes.

On our static LucidLyra-SDG benchmark, LucidLyra improves semantic mIoU and class-agnostic instance mTrackPR over the strongest comparable baseline by 31.7 and 9.8 percentage points, while maintaining reconstruction fidelity, with a modest PSNR gain of 0.53 dB over Lyra. More broadly, this formulation turns pretrained video generation into a scalable source of supervision for structured 3D learning, opening a path from renderable but unstructured outputs to queryable 3D worlds that downstream systems can reason about and interact with. In summary, our contributions are:

- We introduce a unified generative 3D model that produces an explicit Gaussian scene representation with geometry, appearance, open-vocabulary semantic, and class-agnostic instance features from a single image in a feed-forward fashion.
- We develop a multi-task distillation strategy that transfers 2D generative, semantic, and object-level knowledge through aligned generated views into a consistent 3D representation. This enables task-specific training without captured multi-view data or manual 3D annotations, and avoids per-scene optimization at inference time.
- We jointly evaluate reconstruction and understanding on static and dynamic scenes. LucidLyra consistently outperforms two-stage baselines across optimization-based, mask-lifting, and feed-forward 3DGS segmentation methods, while maintaining novel-view reconstruction fidelity on static scenes with a small reconstruction trade-off on dynamic scenes. We further demonstrate downstream applications including robotic simulation, 3D scene editing, and 3D asset mining.

2 Related Work

From observed reconstruction to generative scene construction. Neural radiance fields (Mildenhall et al., 2020) and 3D Gaussian splatting (3DGS) (Kerbl et al., 2023) fit per-scene representations, while feed-forward methods amortize reconstruction across scenes to predict 3D Gaussians for novel-view synthesis (Charatan et al., 2024; Zhang et al., 2024) or estimate dense depth and camera parameters (Wang et al., 2025b; Lin et al., 2026; Burzio et al., 2026). Dynamic formulations underwent a similar evolution from per-sequence fitting (Yang et al., 2024b;a) to feed-forward estimation (Liang et al., 2025b; Xu et al., 2025; Kim et al., 2026), yet all remain bounded by observed multi-view coverage. Generative methods relax this constraint by synthesizing unobserved content. Early works distill diffusion priors into per-scene optimizations or iterative view inpaintings (Poole et al., 2023; Höllein et al., 2023). Multi-view generators synthesize dense video trajectories for downstream reconstruction (Gao et al., 2024; Schneider et al., 2025), with later extensions incorporating panoramic scaffolds, geometric alignment, or long-horizon expansion (Wang et al., 2025a; 2026; Höllein & Nießner, 2026; Shen et al., 2026; Lin et al., 2026; Tian et al., 2026b). However, generation and 3D reconstruction remain decoupled into disparate stages.

A closer line to our setting couples generation directly with feed-forward 3D prediction. Recent methods decode video latents into Gaussians (Liang et al., 2025a), embed Gaussian feature fields within video diffusion (Schwarz et al., 2025), directly generate explicit 3D representations (Szymanowicz et al., 2025; Shabanov et al., 2026), or align generative image, video, and 3D latent spaces (Li et al., 2026e; Huang et al., 2026; Gao et al., 2026). Lyra (Bahmani et al., 2026) self-distills a feed-forward Gaussian decoder from a camera-controlled video foundation model (Ren et al., 2025; NVIDIA, 2025), using synthetic multi-view video trajectories as both the latent input and the supervision target for static and dynamic 3D synthesis. We build upon Lyra's scalable, unannotated latent-to-Gaussian formulation, extending its feed-forward output from raw geometry and appearance to joint language-grounded semantics and class-agnostic instances.

From per-scene optimization to amortized 3D scene understanding. Early works predict semantics and instances from point clouds (Qi et al., 2017a;b; Choy et al., 2019; Jiang et al., 2020), while subsequent methods lift open-vocabulary 2D foundation features onto reconstructed geometry (Peng et al., 2023; Jatavallabhula et al., 2023; Takmaz et al., 2023) or embed language fields into neural representations (Zhi et al., 2021; Kundu et al., 2022; Siddiqui et al., 2023; Kerr et al., 2023). For 3DGS, several approaches optimize per-Gaussian language fields, aggregate 2D foundation features, or lift multi-view masks into object-aware representations (Li et al., 2025b; Xiong et al., 2026; Zhu

et al., 2025; Qin et al., 2024; Tian et al., 2025; Kim et al., 2025; Cheng et al., 2025; Ye et al., 2024; Wu et al., 2024; Shen et al., 2024; Xie et al., 2025). Feed-forward methods amortize understanding across scenes, either by encoding pre-existing 3DGS reconstructions (Li et al., 2026g) or by jointly predicting geometry alongside semantic, panoptic, or instance structure from multi-view inputs (Li et al., 2026c; Zhu et al., 2026; Jiang et al., 2026a; Tian et al., 2026a; Li et al., 2025d; 2026a; Koch et al., 2026; Li et al., 2025a; Sheng et al., 2025; Sun et al., 2026; Liu et al., 2026; Yoo et al., 2026; Lee et al., 2026; Zhang et al., 2026). Generative understanding approaches synthesize observations for language-aware reconstruction, modular scene assembly, or driving-specific world models (Liu et al., 2025; Huang et al., 2025; Yin et al., 2026; Deng et al., 2026). Yet, these formulations remain confined to domain-specific priors or assume multiple posed observations.

Existing approaches follow two main paradigms: augmenting a scene-specific reconstruction or jointly inferring geometry and segmentation from multi-view inputs. The former requires a pre-existing reconstruction, precluding single-view generation, while post-hoc lifting decouples semantics from geometry, leaving fusion vulnerable to cross-view inconsistencies. Crucially, both rely on an expensive preprocessing stage with a 2D vision foundation model. Feed-forward approaches jointly predict geometry and segmentation, but train on captured multi-view datasets with narrow domain coverage limiting generalization. LucidLyra instead unifies scene understanding with generative reconstruction. Conditioned on a single image, its amortized latent-to-Gaussian decoder jointly predicts geometry, appearance, language-aligned semantic features, and class-agnostic instance features in a single feed-forward pass, without prior 3D reconstructions, per-scene optimization, or post-hoc lifting. Distilling frozen foundation models on generated video eliminates the need for captured multi-view data or manual 3D annotations.

3 Semantic- and Instance-Aware Generative 3D Reconstruction

Given an input image $\mathbf{I}_0$, LucidLyra generates a queryable 3D Gaussian Splatting (3DGS) scene representation whose primitives jointly encode geometry, appearance, language-aligned semantics, and class-agnostic instance identity. Rather than supervising reconstruction alone as in Lyra (Bahmani et al., 2026), we use the multi-view outputs of a camera-controlled video generator to construct aligned 2D semantic and instance targets via frozen foundation models and distill them into an explicit 3D representation via an amortized feed-forward decoder (Fig. 2).

3.1 Preliminaries: Lyra Generative 3D Reconstruction

Lyra (Bahmani et al., 2026) amortizes 3D reconstruction by mapping camera-controlled video latents directly into 3D Gaussians. Given an image $\mathbf{I}_0$ and V camera trajectories $\mathcal{C} = (\mathcal{C}^{(1)}, \dots, \mathcal{C}^{(V)})$, a video diffusion model $\mathcal{V}$ generates multi-view latents $\mathbf{Z} = (\mathbf{Z}^{(1)}, \dots, \mathbf{Z}^{(V)})$, while camera rays are mapped to Plücker encodings $\mathbf{E} \equiv \mathbf{E}(\mathcal{C})$. A feed-forward decoder D_{3DGS} predicts a 3DGS scene representation from these inputs:

$$\mathcal{G} = D^{\text{Lyra}}_{\text{3DGS}}(\mathbf{Z}, \mathbf{E}) = \{g_i\}_{i=1}^{N}, \qquad g_i = (\boldsymbol{\mu}_i, \mathbf{s}_i, \mathbf{q}_i, \alpha_i, \mathbf{c}_i), \tag{1}$$

where each of N Gaussian primitives g_i parameterizes position $\boldsymbol{\mu}_i$, scale $\mathbf{s}_i$, rotation quaternion $\mathbf{q}_i$, opacity α_i, and color $\mathbf{c}_i$, renderable via differentiable splatting $\mathcal{R}$. The predicted Gaussians are aligned to the source-ray grid of the generated views across all trajectories.

To learn this mapping without 3D supervision, Lyra compares two decodings of $\mathbf{Z}$: for target camera $\mathcal{C}_k$, a frozen VAE decoder outputs a teacher image $\mathbf{I}_k = D_{\text{RGB}}(\mathbf{Z}_k)$, supervising the rendered prediction $\hat{\mathbf{I}}_k = \mathcal{R}(\mathcal{G}, \mathcal{C}_k)$ alongside depth targets. At inference time, sampled latents $\mathbf{Z}$ yield Gaussians encoding only geometry and appearance.

3.2 Semantic and Instance Gaussian Decoder

To equip Gaussians with semantic and instance awareness, we introduce additional conditioning using vision features of the input image and dedicated output heads. First, we supplement the video model latents $\mathbf{Z}$ with dense input image features $\mathbf{F}_0$ extracted with C-RADIOv3 (Ranzinger et al., 2024; Heinrich et al., 2025; Ranzinger et al., 2025). We align $\mathbf{F}_0$ with $\mathbf{Z}$ through a learned linear projection to the latent feature dimension. Spatial interpolation matches $\mathbf{F}_0$ to the latent

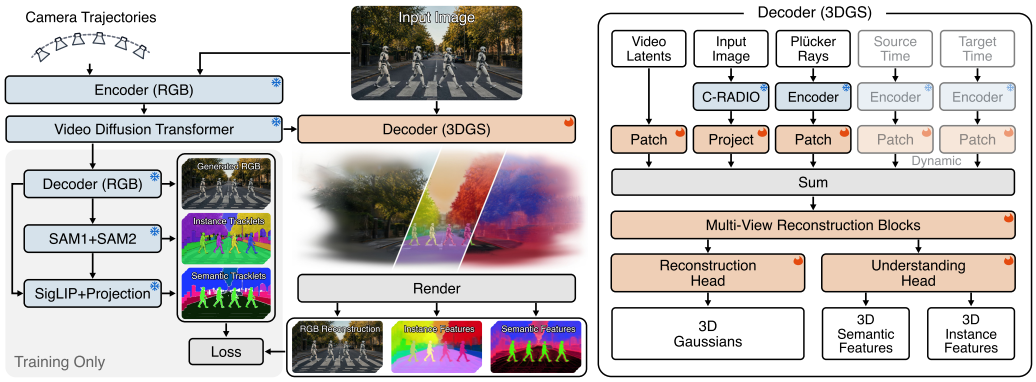


Figure 2: **LucidLyra framework.** A frozen camera-controlled video model produces multi-view latents. Its RGB decoder and frozen 2D foundation models construct reconstruction, semantic, and instance targets. The targets supervise our Gaussian decoder which maps the latents, camera rays, and input image features to a shared queryable 3D scene. The right panel details our 3DGS decoder. Source- and target-time inputs (grayed out) are used only for the dynamic extension. Snowflakes and flames denote frozen and trainable components, respectively.

token grid, and broadcasting replicates it across trajectories and latent time. Each latent-time slice is then modulated by a learned per-channel weight and a learned scalar gate before being added to the video and ray tokens. Next, the 3DGS decoder's shared trunk fuses the multi-view tokens into features $\mathbf{H} = \{\mathbf{h}_i\}_{i=1}^{N}$, where each feature corresponds to a source ray on the multi-view video latent grid. Two parallel heads map these shared features to complementary Gaussian attributes: the reconstruction head D_{rec} predicts geometry and appearance parameters g_i, while the understanding head D_{under} jointly predicts a semantic descriptor $\mathbf{f}_i^{\text{sem}} \in \mathbb{R}^{128}$, an object-level instance embedding $\mathbf{e}_i^{\text{inst}} \in \mathbb{R}^{12}$, and a part-level embedding $\mathbf{e}_i^{\text{part}} \in \mathbb{R}^{4}$:

$$\widetilde{\mathcal{G}} = D_{\text{3DGS}}(\mathbf{Z}, \mathbf{E}, \mathbf{F}_0) = \{\tilde{g}_i\}_{i=1}^{N}, \qquad \tilde{g}_i = [D_{\text{rec}}(\mathbf{h}_i) \parallel D_{\text{under}}(\mathbf{h}_i)]. \tag{2}$$

The shared 3DGS decoder trunk and each task-specific head consist of one Transformer (Vaswani et al., 2017) layer followed by seven Mamba-2 (Dao & Gu, 2024) layers. Operating synchronously on the shared token grid attaches understanding fields directly to the co-located Gaussian g_i, bypassing post-hoc 2D-to-3D association. For camera $\mathcal{C}_k$, a differentiable renderer $\mathcal{R}(\widetilde{\mathcal{G}}, \mathcal{C}_k)$ outputs rendered RGB $\hat{\mathbf{I}}_k$, semantic maps $\hat{\mathbf{F}}_k^{\text{sem}}$, and instance maps $\hat{\mathbf{F}}_k^{\text{inst}}$ via volumetric α-blending, and depth $\hat{\mathbf{D}}_k$ by accumulating primitive center depths.

3.3 Multi-Task 2D Feature Distillation

We supervise the understanding fields using decoded video frames $\mathbf{I}_k$ and frozen 2D vision foundation models—class-agnostic segmentation, video tracking, and vision-language foundation models—generalizing RGB self-distillation to multi-task 2D feature distillation without 3D labels. To resolve cross-view feature jitter and mask inconsistency across imagined views, we first establish cross-view object tracks to associate observations and pool coherent semantic targets.

Aligned instance targets. To associate objects across trajectories, we run SAM's automatic mask generator on the shared first frame $\mathbf{I}_0$, filter redundancies by coverage, and use the surviving masks to seed independent SAM2 propagations along each trajectory (Kirillov et al., 2023; Ravi et al., 2025). The resulting class-agnostic instance tracklets Y_k^{inst} preserve trajectory visibility while linking observations from the same initial object. Regions disoccluded late in a trajectory lacking a first-frame seed are excluded from instance supervision.

Track-consistent semantic targets. The instance tracklets also let us aggregate semantic evidence across views. We use RADSeg (Alama et al., 2026) to extract dense features aligned with the SigLIP2 (Tschannen et al., 2025) image–text space for every decoded frame $\mathbf{I}_k$. Storing its native 1152-dimensional descriptor on every Gaussian is computationally prohibitive, therefore we compress this space with a projection network $P_{\text{VL}} : \mathbb{R}^{1152} \to \mathbb{R}^{128}$ for both image and text embeddings. The projector is trained over frozen SigLIP2 image and text embeddings, using the vanilla SigLIP

training objective (Zhai et al., 2023) on the LAION-2B (Schuhmann et al., 2022) dataset and remains frozen afterwards. Within each camera trajectory, we pool projected features over all pixels and frames belonging to each SAM track and assign the resulting centroid to every track pixel. This yields semantic targets $\mathbf{Y}_k^{\text{sem}}$, where track pooling suppresses view-dependent feature jitter.

Joint multi-task objective. We initialize the shared trunk and reconstruction head D_{rec} from pretrained Lyra and initialize D_{under} from scratch, training the entire Gaussian decoder while keeping all 2D teacher models and P_{VL} frozen. At target camera $\mathcal{C}_k$, rendered predictions $(\hat{\mathbf{I}}_k, \hat{\mathbf{D}}_k, \hat{\mathbf{F}}_k^{\text{sem}}, \hat{\mathbf{F}}_k^{\text{inst}})$ are supervised against targets $(\mathbf{I}_k, \mathbf{D}_k, \mathbf{Y}_k^{\text{sem}}, Y_k^{\text{inst}})$ via

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{sem}} + \lambda_{\text{inst}}\mathcal{L}_{\text{inst}}. \tag{3}$$

$\mathcal{L}_{\text{rec}}$ mirrors Lyra, combining photometric ℓ_1, LPIPS, depth, and opacity losses to preserve geometry during feature grounding. Schedules and hyperparameters are detailed in Appendix A.

Instance distillation. We learn a class-agnostic instance representation via supervised contrastive learning (Khosla et al., 2020) over tracked pixels across views. For a sample $u \in \mathcal{S}$ at pixel p_u in view k_u with normalized embedding $\bar{\mathbf{e}}_u = \text{norm}_2(\hat{\mathbf{F}}_{k_u}^{\text{inst}}(p_u))$, we define positive matches sharing track ID as $\mathcal{P}(u) = \{v \in \mathcal{S} \setminus \{u\} \mid Y_{k_v}^{\text{inst}}(p_v) = Y_{k_u}^{\text{inst}}(p_u)\}$. Over valid anchors $\mathcal{U} = \{u \in \mathcal{S} \mid |\mathcal{P}(u)| > 0\}$, we optimize

$$\mathcal{L}_{\text{inst}} = -\frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{1}{|\mathcal{P}(u)|}\sum_{v\in\mathcal{P}(u)}\log\frac{\exp(\bar{\mathbf{e}}_u^\top\bar{\mathbf{e}}_v/\tau_{\text{inst}})}{\sum_{a\in\mathcal{S}\setminus\{u\}}\exp(\bar{\mathbf{e}}_u^\top\bar{\mathbf{e}}_a/\tau_{\text{inst}})}, \tag{4}$$

with temperature $\tau_{\text{inst}} > 0$. This pulls rendered embeddings of the same tracked object together across views while repelling different objects, clustering the underlying 3D Gaussians into distinct instance fields. For part-level supervision, we apply the same objective in parallel to the part embeddings, defining positive and negative pairs from sub-object tracklet IDs within each object tracklet.

Semantic distillation. We supervise rendered semantic features with pooled teacher descriptors. For rendered feature $\hat{\mathbf{f}} \equiv \hat{\mathbf{F}}_k^{\text{sem}}(p)$ and target $\mathbf{y} \equiv \mathbf{Y}_k^{\text{sem}}(p)$ with ℓ_2-normalized counterparts $\bar{\hat{\mathbf{f}}}$ and $\bar{\mathbf{y}}$, we define the pointwise loss

$$\ell_{\text{sem}}(\hat{\mathbf{f}}, \mathbf{y}) = -\frac{1}{2}\log\left(\frac{1+\bar{\hat{\mathbf{f}}}^\top\bar{\mathbf{y}}}{2}+\epsilon_\ell\right) + \frac{1}{2d_{\text{sem}}}\left\|\bar{\hat{\mathbf{f}}}-\bar{\mathbf{y}}\right\|_1, \tag{5}$$

with $d_{\text{sem}} = 128$. The cosine term aligns predictions with the SigLIP2 image–text space, while the ℓ_1 term refines coordinate-level precision. Averaging over rendered pixels yields $\mathcal{L}_{\text{sem}}^{\text{2D}}$. Rendered feature losses constrain a blend of several Gaussian features, rather than each primitive individually. We therefore additionally supervise each primitive directly: each Gaussian $\tilde{g}_i$ originates from an input ray (k_i, p_i), allowing us to penalize discrepancy against teacher target $\mathbf{Y}_{k_i}^{\text{sem}}(p_i)$ as $\mathcal{L}_{\text{sem}}^{\text{G}}$, yielding the total semantic loss $\mathcal{L}_{\text{sem}} = \lambda_{\text{sem}}^{\text{2D}}\mathcal{L}_{\text{sem}}^{\text{2D}} + \lambda_{\text{sem}}^{\text{G}}\mathcal{L}_{\text{sem}}^{\text{G}}$.

3.4 Inference: Querying the Generated Scene

Given generated video latents, our Gaussian decoder predicts the scene $\widetilde{\mathcal{G}}$ in a single forward pass. Its per-Gaussian semantic and instance features enable open-vocabulary queries and object grouping directly in 3D, without using the distillation teachers or lifting predictions from 2D views.

Open-vocabulary semantics. Given categorical text queries $\mathcal{Q} = \{s_c\}_{c=1}^C$, the frozen SigLIP2 text encoder Φ_{T} and projection network P_{VL} yield normalized embeddings $\bar{\mathbf{t}}_c = \text{norm}_2(P_{\text{VL}}(\Phi_{\text{T}}(s_c)))$. Each Gaussian $\tilde{g}_i$ is assigned to the category c^* that maximizes the cosine similarity with its primitive descriptor:

$$c^* = \arg\max_{c\in\{1,\ldots,C\}} \text{norm}_2(\mathbf{f}_i^{\text{sem}})^\top\bar{\mathbf{t}}_c. \tag{6}$$

Querying new vocabularies needs only text–Gaussian dot products without regeneration.

Instance grouping. Within each predicted semantic class, we spatially smooth instance embeddings and cluster high-opacity Gaussians via HDBSCAN (Campello et al., 2013). Unclustered Gaussians inherit instance IDs from their nearest neighbor in joint embedding and 3D space. For the part-level instance segmentation, we apply the same technique within each object cluster (cf. Appendix A.3).

LucidLyra+. Our optional semantic refinement leaves geometry and appearance unchanged. It uses a single RADSeg+ open-vocabulary prediction of $\mathbf{I}_0$ as a 2D anchor, obtained at minimal overhead since RADSeg+ uses the same C-RADIO features. Our ray-aligned decoder grid links each anchor pixel to its corresponding Gaussian. For every well-supported anchor, we average the semantic features of its Gaussians into a prototype, and relabel any Gaussian whose feature is close enough to that prototype. All remaining Gaussians keep their original semantic labels. We then apply the same instance grouping.

3.5 Extension to Dynamic Scenes

For monocular video inputs, we extend LucidLyra to dynamic scenes by independently decoding a time-conditioned 3DGS state following Lyra's bullet-time formulation (Bahmani et al., 2026; Liang et al., 2025b). The same video model provides multi-view latents preserving input motion; the decoder predicts the full scene state at time τ, conditioned on source/target time encodings and query-time C-RADIO input-frame features $\mathbf{F}_\tau$:

$$\widetilde{\mathcal{G}}_\tau = D^{\mathrm{dyn}}_{\mathrm{3DGS}}(\mathbf{Z}, \mathbf{E}, \mathbf{F}_\tau, \mathbf{T}^{\mathrm{src}}, \mathbf{T}^{\mathrm{tgt}}_\tau). \tag{7}$$

Initialized from the static model, the dynamic decoder trains with the same multi-task losses on τ-synchronized views. Dynamic scenes can only be supervised at the exact queried time, due to the changing scene. Following Bahmani et al. (2026), we create six additional trajectories by processing the time-reversed input video and flipping it back. Each target frame is then paired with a same-time observation from a different viewpoint increasing multi-view supervision. Since each query time yields an independently decoded Gaussian state, we implement temporal instance tracking via an online prototype codebook inspired by IGGT4D (Zou et al., 2026). At each query time, we represent each cluster by its normalized mean object embedding and greedily match cluster prototypes to codebook entries one-to-one above a cosine-similarity threshold. Matched clusters inherit persistent IDs and update the stored prototypes through averaging weighted by Gaussian count, while unmatched clusters initialize new IDs.

4 Experiments

We evaluate LucidLyra's joint generative 3D reconstruction and scene understanding on synthetic and real-image-conditioned scenes, followed by an analysis of its key components. Dynamic experiments, additional quantitative and qualitative results, downstream applications, and implementation details are provided in the appendix and at https://anonlucidlyra.github.io.

4.1 Experimental Setup

Datasets. We augment Lyra-SDG (Bahmani et al., 2026) with semantic and instance teacher targets for training (Section 3.3) and annotations for evaluation. Lyra-SDG uses GEN3C (Ren et al., 2025) to expand images and videos generated from diverse text prompts into multiple camera trajectories per scene. Its $59\,031$ static and 7378 dynamic scenes contain RGB videos, camera poses, depth maps, and video latents. We preprocess teacher targets for all 53 million frames. A single image admits multiple plausible 3D completions. We evaluate agreement with fixed generated trajectories and their annotations. Our three curated 25-scene benchmarks use propagated reference annotations with human review and refinement: *LucidLyra-SDG* and *LucidLyra-SDG Dynamic* cover static and dynamic generated scenes, while *LucidLyra-ADE20K* uses real ADE20K validation images (Zhou et al., 2019) to condition generated completions. All data and benchmarks will be released.

For qualitative results, we generate input images with GPT Image or FLUX (OpenAI, 2026; Black Forest Labs, 2024). For dynamic examples, we extend these to videos using Cosmos 3 (NVIDIA, 2026b). We find that our method is agnostic to the choice of image-generation model.

Evaluation. Following Lyra (Bahmani et al., 2026), we report PSNR and LPIPS (Zhang et al., 2018) for reconstruction. Following Li et al. (2025b); Xiong et al. (2026); Li et al. (2026g;c), we evaluate semantic segmentation using mIoU and mAcc over ADE20K classes present in each reference scene. Building on IGGT and video instance segmentation (Li et al., 2026c; Yang et al., 2019), we evaluate cross-view instance consistency using the mean track precision–recall product (mTrackPR) and $\mathrm{TrackPR}_{25}$ at IoU 0.25 within ground-truth thing regions (Appendix A.3).

Table 1: **Static evaluation on LucidLyra-SDG and LucidLyra-ADE20K.** We report PSNR (dB), LPIPS (–), and mIoU, mAcc, mTrackPR, and TrackPR_{25} (all in %). †: uses video model RGB output.

Method	LucidLyra-SDG *Static*						LucidLyra-ADE20K					
	Reconstruction		*Semantics*		*Instance*		*Reconstruction*		*Semantics*		*Instance*	
	PSNR↑	LPIPS↓	mIoU↑	mAcc↑	mTrack PR↑	Track PR_{25}↑	PSNR↑	LPIPS↓	mIoU↑	mAcc↑	mTrack PR↑	Track PR_{25}↑
Lyra + Lifted Teachers	24.49	0.211	60.5	75.3	22.8	33.2	24.59	0.217	50.1	64.5	17.3	26.8
Lyra	24.49	0.211	–	–	–	–	24.59	0.217	–	–	–	–
Lyra + LangSplatV2	24.49	0.211	32.4	52.7	–	–	24.59	0.217	37.1	53.7	–	–
Lyra + SFS	24.49	0.211	31.9	49.5	–	–	24.59	0.217	35.2	51.3	–	–
Lyra + Unified-Lift	24.49	0.211	–	–	11.7	17.0	24.59	0.217	–	–	7.5	11.5
Lyra + Chorus	24.49	0.211	11.5	19.3	–	–	24.59	0.217	9.3	16.1	–	–
Lyra† + IGGT	–	–	29.2	38.3	5.2	9.3	–	–	32.7	41.2	9.5	16.6
LucidLyra *(Ours)*	**25.02**	**0.186**	64.1	79.2	21.5	29.3	**24.72**	**0.199**	49.8	65.6	**18.5**	**29.7**
LucidLyra+ *(Ours)*	**25.02**	**0.186**	**67.9**	**81.7**	**22.0**	**31.1**	**24.72**	**0.199**	**53.7**	**71.6**	**18.5**	28.5

Baselines. We benchmark against existing 3D understanding methods combined with Lyra's generative pipeline. *Per-scene optimization*: LangSplatV2, SFS, and Unified-Lift (Li et al., 2025b; Xiong et al., 2026; Zhu et al., 2025) fit feature fields on frozen Lyra Gaussians. *Feed-forward*: Chorus (Li et al., 2026g) processes frozen Lyra Gaussians, while IGGT (Li et al., 2026c) reconstructs geometry and instances directly from generated RGB videos. *References*: Reconstruction-only Lyra, alongside a teacher-lifting reference that directly assigns teacher features and instance IDs to frozen Lyra ray-aligned-Gaussians. The latter contextualizes the quality of our segmentation training signals.

4.2 Evaluating Generative 3D Reconstruction and Understanding

LucidLyra preserves reconstruction fidelity while adding understanding fields (Table 1). Relative to Lyra, PSNR improves by $0.53/0.13$ dB and LPIPS improves by $0.025/0.018$ across benchmarks, confirming that our multi-task training does not degrade novel-view rendering quality.

LucidLyra outperforms all two-stage baselines without per-scene optimization (Table 1) in semantics, achieving $64.1/49.8$ mIoU with synthetic and real-image conditioning, respectively. Plain LucidLyra exceeds the 3D-lifted teacher reference on SDG (64.1 vs. 60.5) and performs comparably on ADE20K (49.8 vs. 50.1). LucidLyra+ input-image 3D anchoring further boosts mIoU to $67.9/53.7$.

For instances, LucidLyra achieves $21.5/18.5$ instance mTrackPR (Table 1), outperforming the compared segmentation baselines in separating individual objects across generated viewpoints. Compared to the lifted teacher reference, mTrackPR is lower on SDG (21.5 vs. 22.8) and higher on ADE20K (18.5 vs. 17.3). LucidLyra+ gives a small SDG mTrackPR gain. Qualitative multi-view segmentations and feature PCAs are visualized in Fig. 3.

On a single NVIDIA A100 80 GB, teacher lifting, LangSplatV2, SFS, and Unified-Lift require 64, 697, 168, and 216 min to segment a single generated scene, respectively. This is dominated by foundation-model preprocessing. Feed-forward methods CHORUS and IGGT require 1.0 and 3.6 min, respectively, while LucidLyra and LucidLyra+ require 2.1 and 2.3 min.

4.3 LucidLyra Analysis

Losses and regularization. Removing C-RADIO conditioning causes the largest drop in understanding: 7.2 mIoU and 6.6 mTrackPR points (Table 2). Direct Gaussian semantic supervision adds 0.7 mIoU and 2.7 mTrackPR points beyond rendered supervision. Reconstruction masking and opacity pruning add 1.3 and 2.1 mTrackPR points, respectively, with small changes in reconstruction quality. Part-level supervision slightly improves mIoU and enables a more fine-grained scene interaction. The full model also retains reconstruction quality, slightly improving PSNR (25.02 vs. 24.88) and LPIPS (0.186 vs. 0.189) over the reconstruction-only control run.

Feature dimensions. Reducing the semantic feature dimension from 128 to 64 lowers mIoU by 1.4 points and mTrackPR by 1.0 point. A 256-dimensional representation exceeds the GPU memory budget. The chosen 128D achieves the best accuracy–memory trade-off among the tested settings.

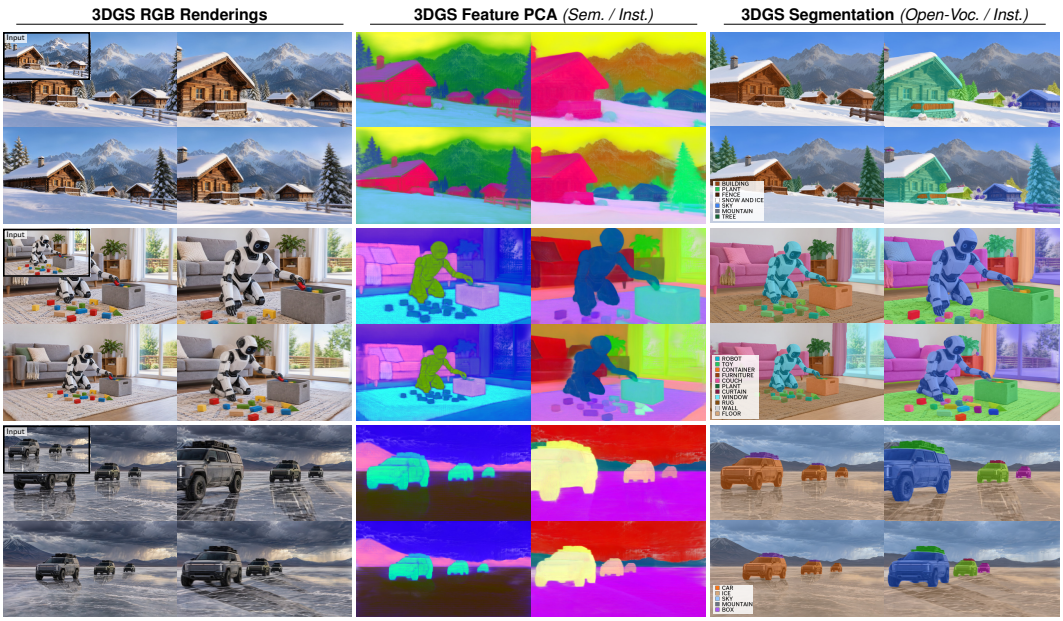


Figure 3: **LucidLyra qualitative results.** We visualize input- and novel-view 3DGS renderings across RGB, semantic/instance feature PCAs, and open-vocabulary/instance segmentation.

The 12-dimensional object-level instance representation achieves the highest mTrackPR among the tested settings: 21.5, compared with 19.3 at 8 dimensions and 18.1 at 16 dimensions. Semantic mIoU is unchanged at 8 dimensions and 1.0 point lower at 16 dimensions.

Shared 3DGS decoder layers. Increasing the shared decoder depth from eight to twelve layers degrades reconstruction and understanding. Four shared layers remain competitive, while eight shared layers achieve the highest mTrackPR among the tested depths.

4.4 Downstream Applications

LucidLyra's structured, queryable scenes directly enable downstream tasks including robotic simulation, 3D scene editing, and 3D asset mining (details and qualitative results in Appendix D).

Table 2: **Ablations on LucidLyra-SDG.** We report PSNR (dB), LPIPS (–), and mIoU, mAcc, mTrackPR, and TrackPR25 (all in %).

Method	*Reconstruction*		*Semantics*		*Instance*	
	PSNR↑	LPIPS↓	mIoU↑	mAcc↑	mTrack PR↑	Track PR_{25}↑
LucidLyra *(Ours)*	25.02	0.186	64.1	79.2	21.5	29.3
Loss						
w/o 3D Loss	24.89	0.190	63.4	77.0	18.8	26.2
w/o C-RADIO	24.77	0.190	56.9	70.2	14.9	24.5
w/o part-level supervision	25.00	0.187	63.8	79.0	21.0	29.6
w/o opacity pruning	25.03	0.187	63.7	78.7	19.4	27.7
w/o ignore mask recon.	24.84	0.189	64.0	78.8	20.2	28.7
w/o LucidLyra	24.88	0.189	–	–	–	–
Semantic Feature Dim.						
64	24.83	0.189	62.7	76.2	20.5	30.2
256			OOM			
Instance Feature Dim.						
8	24.88	0.188	64.1	78.6	19.3	28.7
16	24.98	0.187	63.1	77.6	18.1	27.0
Shared 3DGS Decoder Layers						
4	25.03	0.184	64.3	79.1	21.0	29.8
12	24.83	0.189	63.6	78.2	18.6	27.9

5 Conclusion

We presented LucidLyra, which turns generated scenes into queryable 3D worlds by distilling generative and perceptual knowledge into a feed-forward Gaussian decoder. A camera-controlled video model supplies both latent representations and training data, enabling joint reconstruction and understanding without captured multi-view data, manual 3D annotations, or per-scene optimization. Experiments demonstrate improved semantic and instance understanding over the compared two-stage baselines while preserving Lyra's reconstruction quality. Furthermore, we demonstrate initial results on dynamic scenes and applications in physical AI simulation, scene editing, and 3D asset extraction. Future work could scale our approach by accelerated video generation (Li et al., 2026f) with larger-scale joint training for generation and scene understanding (Gabeur et al., 2026).

6 STATEMENTS

6.1 REPRODUCIBILITY STATEMENT

To support reproducibility and further research on generative 3D reconstruction and scene understanding, we will publicly release the training and inference code and pretrained weights for LucidLyra upon acceptance. For both the static and dynamic training sets, we will provide the raw preprocessing outputs and the final semantic and instance supervision targets. We will also release the annotations for all three evaluation benchmarks: LucidLyra-SDG, LucidLyra-ADE20K, and LucidLyra-SDG Dynamic.

6.2 ETHICS STATEMENT

Like other generative AI technologies, our method carries risks of misuse, including the creation of misleading synthetic 3D content. Semantics- and instance-aware representations may also make it easier to identify, select, or alter particular objects within generated scenes. Nevertheless, the primary goal of this research is to advance simulation, robotics, and embodied AI by developing scalable and controllable tools for generating semantically grounded 3D data.

We expect the main benefits of this work to arise from the following capabilities and insights: *(I)* **More useful scene representations** for embodied agents. Our method augments generated 3D scenes with open-vocabulary semantic and instance fields, allowing objects and regions to be queried and selected consistently across viewpoints. Such representations may support the development and evaluation of agents that must perceive, navigate, and interact with complex environments. Our current method does not model physics or establish safe closed-loop behavior, and should therefore be regarded as a step toward, rather than a complete solution for, embodied simulation. *(II)* **Improved scalability for data engines.** Our approach reduces reliance on captured multi-view data, manual 3D annotations, and per-scene optimization. Instead, it learns semantic and instance-aware scene representations from generated trajectories and supervision provided by pretrained 2D models. This design may facilitate the construction of customizable 3D datasets for research in robotics, reinforcement learning, and simulation. *(III)* **Better technical understanding.** As an academic study, this work provides a case study of how semantic conditioning, instance supervision, and generative reconstruction interact within a shared 3D representation. We expect these findings to inform future research at the intersection of computer vision, graphics, and embodied AI.

We caution against deceptive or otherwise harmful uses of this technology. The model may also inherit errors and biases from its training data and pretrained teachers. Responsible deployment should therefore include provenance tracking, disclosure of generated or edited content, documentation of datasets and known limitations, and careful evaluation before use in consequential settings. Although such safeguards cannot eliminate misuse, they can improve transparency and accountability as generative 3D models are applied in research and society.

6.3 AI DISCLOSURE STATEMENT

Generative AI tools were used throughout the entire research workflow, including for technical brainstorming, software development, debugging, experiment organization, and analysis. Their role was consistently assistive: the authors set the objectives and constraints, worked interactively with the tools, inspected their outputs, and made all scientific and technical decisions. No AI-assisted contribution was incorporated into the project without author review and validation. During manuscript preparation, generative AI tools assisted with structuring and rephrasing text for clarity. The authors reviewed and revised the entire final manuscript and take full responsibility for its content.

REFERENCES

Omar Alama, Darshil Jariwala, Avigyan Bhattacharya, Seungchan Kim, Wenshan Wang, and Sebastian Scherer. RADSeg: Unleashing parameter and compute efficient zero-shot open-vocabulary segmentation using agglomerative models. In *CVPR Findings*, pp. 9294–9304, 2026.

Sherwin Bahmani, Tianchang Shen, Jiawei Ren, Jiahui Huang, Yifeng Jiang, Haithem Turki, Andrea Tagliasacchi, David B. Lindell, Zan Gojcic, Sanja Fidler, Huan Ling, Jun Gao, and Xuanchi Ren. Lyra: Generative 3D scene reconstruction via video diffusion model self-distillation. In *ICLR*, 2026.

Black Forest Labs. FLUX. `https://github.com/black-forest-labs/flux`, 2024.

Alessandro Burzio, Tobias Fischer, Sven Elflein, Qunjie Zhou, Riccardo de Lutio, Jiawei Ren, Jiahui Huang, Shengyu Huang, Marc Pollefeys, Laura Leal-Taixé, Zan Gojcic, and Haithem Turki. Déjà View: Looping transformers for multi-view 3D reconstruction. *arXiv:2605.30215 [cs.CV]*, 2026.

Ricardo J. G. B. Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *PAKDD*, pp. 160–172, 2013.

David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelSplat: 3D Gaussian splats from image pairs for scalable generalizable 3D reconstruction. In *CVPR*, pp. 19457–19467, 2024.

Jiahuan Cheng, Jan-Nico Zaech, Luc Van Gool, and Danda Pani Paudel. Occam's LGS: An efficient approach for language Gaussian splatting. In *BMVC*, 2025.

Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4D spatio-temporal ConvNets: Minkowski convolutional neural networks. In *CVPR*, pp. 3075–3084, 2019.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *ICML*, pp. 10041–10071, 2024.

Tianchen Deng, Xuefeng Chen, Yi Chen, Qu Chen, Yuyao Xu, Lijin Yang, Le Xu, Yu Zhang, Bo Zhang, Wuxiong Huang, and Hesheng Wang. GaussianDWM: 3D Gaussian driving world model for unified scene understanding and multi-modal generation. In *CVPR*, 2026.

Gal Fiebelman, Tamir Cohen, Ayellet Morgenstern, Peter Hedman, and Hadar Averbuch-Elor. 4-LEGS: 4D language embedded Gaussian splatting. *Comput. Graph. Forum*, 44(2):e70085, 2025.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T. Barron, Kyle Genova, Nithish Kannen, Sherry Ben, Yandong Li, Mandy Guo, Suhas Yogin, Yiming Gu, Huizhong Chen, Oliver Wang, Saining Xie, Howard Zhou, Kaiming He, Thomas Funkhouser, Jean-Baptiste Alayrac, and Radu Soricut. Image generators are generalist vision learners. *arXiv:2604.20329 [cs.CV]*, 2026.

Ruiqi Gao, Aleksander Hołyński, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul P. Srinivasan, Jonathan T. Barron, and Ben Poole. CAT3D: Create anything in 3D with multi-view diffusion models. In *NeurIPS*, 2024.

Sensen Gao, Zhaoqing Wang, Qihang Cao, Dongdong Yu, Changhu Wang, and Jia-Wang Bian. PixWorld: Unifying 3D scene generation and reconstruction in pixel space. *arXiv:2607.05373 [cs.CV]*, 2026.

Shuting He, Peilin Ji, Yitong Yang, Changshuo Wang, Jiayi Ji, Yinglin Wang, and Henghui Ding. A survey on 3D Gaussian splatting applications: Segmentation, editing, and generation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2026.

Greg Heinrich, Mike Ranzinger, Hongxu Yin, Yao Lu, Jan Kautz, Andrew Tao, Bryan Catanzaro, and Pavlo Molchanov. RADIOv2.5: Improved baselines for agglomerative vision foundation models. In *CVPR*, 2025.

Lukas Höllein and Matthias Nießner. World reconstruction from inconsistent views. In *ECCV*, 2026.

Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2Room: Extracting textured 3D meshes from 2D text-to-image models. In *ICCV*, pp. 7909–7920, 2023.

Yongzhen Hu, Yihui Yang, Haotong Lin, Yifan Wang, Junting Dong, Yifu Deng, Xinyu Zhu, Fan Jia, Hujun Bao, Xiaowei Zhou, and Sida Peng. Split4D: Decomposed 4D scene reconstruction without video segmentation. *ACM Trans. Graph.*, 44(6):215:1–215:15, 2025.

Jiaxin Huang, Yuanbo Yang, Bangbang Yang, Lin Ma, Yuewen Ma, and Yiyi Liao. Gen3R: 3D scene generation meets feed-forward reconstruction. In *CVPR*, 2026.

Rui Huang, Guangyao Zhai, Zuria Bauer, Marc Pollefeys, Federico Tombari, Leonidas [J.] Guibas, Gao Huang, and Francis Engelmann. Video perception models for 3D scene synthesis. In *NeurIPS*, 2025.

Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Ganesh Iyer, Soroush Saryazdi, Tao Chen, Alaa Maalouf, Shuang Li, Nikhil Varma Keetha, Ayush Tewari, Joshua Tenenbaum, Celso de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. ConceptFusion: Open-set multimodal 3D mapping. In *RSS*, 2023.

Shengxiang Ji, Guanjun Wu, Jiemin Fang, Jiazhong Cen, Taoran Yi, Wenyu Liu, Qi Tian, and Xinggang Wang. Segment any 4D Gaussians. *arXiv:2407.04504 [cs.CV]*, 2024.

Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. PointGroup: Dual-set point grouping for 3D instance segmentation. In *CVPR*, pp. 4867–4876, 2020.

Minchao Jiang, Xiaoxuan Ma, Shunyu Jia, Haoru Wang, Zhang Liang, and Wentao Zhu. InstanceSplat: Instance-aware feed-forward 3D Gaussian splatting for scene understanding. *arXiv:2608.07144 [cs.CV]*, 2026a.

Yuheng Jiang, Yiwen Cai, Zihao Wang, Yize Wu, Sicheng Li, Zhuo Su, Shaohui Jiao, and Lan Xu. Director: Instance-aware Gaussian splatting for dynamic scene modeling and understanding. In *ECCV*, 2026b.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–139:14, 2023.

Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language embedded radiance fields. In *ICCV*, pp. 19729–19739, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, pp. 18661–18673, 2020.

Jun-Seong Kim, GeonU Kim, Yu-Ji Kim, Yu-Chiang Frank Wang, Jaesung Choe, and Tae-Hyun Oh. Dr. Splat: Directly referring 3D Gaussian splatting via direct language embedding registration. In *CVPR*, pp. 14137–14146, 2025.

Mungyeom Kim, Minkyeong Jeon, Honggyu An, Jaewoo Jung, Hyuna Ko, Jisang Han, Hyeonseo Yu, Donghwan Shin, Sunghwan Hong, Takuya Narihira, Kazumi Fukuda, Yuki Mitsufuji, and Seungryong Kim. Learning global motion with compact Gaussians for feed-forward 4D reconstruction. *arXiv:2605.31595 [cs.CV]*, 2026.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. In *ICCV*, pp. 4015–4026, 2023.

Sebastian Koch, Johanna Wald, Hidenobu Matsuki, Pedro Hermosilla, Timo Ropinski, and Federico Tombari. Unified semantic transformer for 3D scene understanding. *Trans. Mach. Learn. Res.*, 2026.

Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas J. Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas Funkhouser. Panoptic neural fields: A semantic object-aware neural scene representation. In *CVPR*, pp. 12871–12881, 2022.

Isaac Labe, Noam Issachar, Itai Lang, and Sagie Benaim. DGD: Dynamic 3D Gaussians distillation. In *ECCV*, pp. 361–378, 2024.

Seungjun Lee, Zihan Wang, Yunsong Wang, and Gim Hee Lee. EmbodiedSplat: Online feed-forward semantic 3DGS for open-vocabulary 3D scene understanding. In *CVPR*, pp. 23774–23784, 2026.

Changyang Li, Xueqing Huang, Shin-Fang Chng, Huangying Zhan, Qingan Yan, and Yi Xu. FAST3DIS: Feed-forward anchored scene transformer for 3D instance segmentation. *arXiv:2603.25993 [cs.CV]*, 2026a.

Dong-Yang Li, Wang Zhao, Yuxin Chen, Wenbo Hu, Meng-Hao Guo, Fang-Lue Zhang, Ying Shan, and Shi-Min Hu. Pixal3D: Pixel-aligned 3D generation from images. In *SIGGRAPH*, 2026b.

Hao Li, Zhengyu Zou, Fangfu Liu, Xuanyang Zhang, Fangzhou Hong, Yukang Cao, Yushi Lan, Manyuan Zhang, Gang Yu, Dingwen Zhang, and Ziwei Liu. IGGT: Instance-grounded geometry transformer for semantic 3D reconstruction. In *ICLR*, 2026c.

Qijing Li, Jingxiang Sun, Liang An, Zhaoqi Su, Hongwen Zhang, and Yebin Liu. SemanticSplat: Feed-forward 3D scene understanding with language-aware Gaussian fields. *arXiv:2506.09565 [cs.CV]*, 2025a.

Ruining Li, Yuxin Yao, Chuanxia Zheng, Christian Rupprecht, Joan Lasenby, Shangzhe Wu, and Andrea Vedaldi. Particulate: Feed-forward 3D object articulation. In *CVPR*, pp. 27708–27718, 2026d.

Wanhua Li, Yujie Zhao, Minghan Qin, Yang Liu, Yuanhao Cai, Chuang Gan, and Hanspeter Pfister. LangSplatV2: High-dimensional 3D language Gaussian splatting with 450+ FPS. In *NeurIPS*, 2025b.

Wanhua Li, Renping Zhou, Jiawei Zhou, Yingwei Song, Johannes Herter, Minghan Qin, Gao Huang, and Hanspeter Pfister. 4D LangSplat: 4D language Gaussian splatting via multimodal large language models. In *CVPR*, pp. 22001–22011, 2025c.

Xinyang Li, Tengfei Wang, Zixiao Gu, Shengchuan Zhang, Chunchao Guo, and Liujuan Cao. FlashWorld: High-quality 3D scene generation within seconds. In *ICLR*, 2026e.

Yitong Li, Junsong Chen, Haopeng Li, Haozhe Liu, Jincheng Yu, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Sol Video Inference Engine: Agent-native full-stack acceleration framework for efficient video generation. *arXiv:2606.23743 [cs.CV]*, 2026f.

Yue Li, Qi Ma, Runyi Yang, Huapeng Li, Mengjiao Ma, Bin Ren, Nikola Popovic, Nicu Sebe, Ender Konukoglu, Theo Gevers, Luc Van Gool, Martin R. Oswald, and Danda Pani Paudel. SceneSplat: Gaussian splatting-based scene understanding with vision-language pretraining. In *ICCV*, pp. 4961–4972, 2025d.

Yue Li, Qi Ma, Runyi Yang, Mengjiao Ma, Bin Ren, Nikola Popovic, Nicu Sebe, Theo Gevers, Luc Van Gool, Danda Pani Paudel, and Martin R. Oswald. Chorus: Multi-teacher pretraining for holistic 3D Gaussian scene encoding. In *CVPR*, pp. 21431–21442, 2026g.

Yun-Jin Li, Mariia Gladkova, Yan Xia, and Daniel Cremers. TRASE: Tracking-free 4D segmentation and editing. In *3DV*, pp. 1670–1679, 2026h.

Hanwen Liang, Junli Cao, Vidit Goel, Guocheng Qian, Sergei Korolev, Demetri Terzopoulos, Konstantinos N. Plataniotis, Sergey Tulyakov, and Jian Ren. Wonderland: Navigating 3D scenes from a single image. In *CVPR*, pp. 798–810, 2025a.

Hanxue Liang, Jiawei Ren, Ashkan Mirzaei, Antonio Torralba, Ziwei Liu, Igor Gilitschenski, Sanja Fidler, Cengiz Öztireli, Huan Ling, Zan Gojcic, and Jiahui Huang. Feed-forward bullet-time reconstruction of dynamic scenes from monocular videos. In *NeurIPS*, 2025b.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Yang Zhao, Sida Peng, Hengkai Guo, Xiaowei Zhou, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth Anything 3: Recovering the visual space from any views. In *ICLR*, 2026.

Fangfu Liu, Hao Li, Jiawei Chi, Hanyang Wang, Minghui Yang, Fudong Wang, and Yueqi Duan. LangScene-X: Reconstruct generalizable 3D language-embedded scenes with TriMap video diffusion. In *ICCV*, 2025.

Yilong Liu, Wanhua Li, Chen Zhu-Tian, and Hanspeter Pfister. LangFlash: Feed-forward 3D language Gaussian splatting from sparse unposed images. In *CVPR Findings*, 2026.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, pp. 405–421, 2020.

NVIDIA. Cosmos world foundation model platform for physical AI. *arXiv:2501.03575 [cs.CV]*, 2025.

NVIDIA. Isaac Sim. `https://github.com/isaac-sim/IsaacSim`, 2026a.

NVIDIA. Cosmos 3: Omnimodal world models for physical AI. *arXiv:2606.02800 [cs.CV]*, 2026b.

OpenAI. ChatGPT Images 2.5 system card. `https://deploymentsafety.openai.com/chatgpt-images-2-5`, 2026.

Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D scene understanding with open vocabularies. In *CVPR*, pp. 815–824, 2023.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D diffusion. In *ICLR*, 2023.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, pp. 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 *NeurIPS*, 2017b.

Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. LangSplat: 3D language Gaussian splatting. In *CVPR*, pp. 20051–20060, 2024.

Zhicheng Qiu, Jiarui Meng, Tong-an Luo, Yican Huang, Xuan Feng, Xuanfu Li, and Zhan Xu. SLARM: Streaming and language-aligned reconstruction model for dynamic scenes. In *CVPR*, pp. 29023–29034, 2026.

Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. AM-RADIO: Agglomerative vision foundation model reduce all domains into one. In *CVPR*, pp. 12490–12500, 2024.

Mike Ranzinger, Greg Heinrich, Pavlo Molchanov, Bryan Catanzaro, and Andrew Tao. FeatSharp: Your vision model features, sharper. In *ICML*, 2025.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloé Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross B. Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *ICLR*, 2025.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. GEN3C: 3D-informed world-consistent video generation with precise camera control. In *CVPR*, pp. 6121–6132, 2025.

Manuel-Andreas Schneider, Lukas Höllein, and Matthias Nießner. WorldExplorer: Towards generating fully navigable 3D scenes. In *SIGGRAPH Asia*, 2025.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W. Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R. Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *NeurIPS Datasets & Benchmarks*, 2022.

Katja Schwarz, Norman Müller, and Peter Kontschieder. Generative Gaussian splatting: Generating 3D scenes with video diffusion priors. In *ICCV*, pp. 27510–27520, 2025.

Ahan Shabanov, Peter Hedman, Ethan Weber, Zhengqin Li, Denis Rozumny, Gael Le Lan, Naina Dhingra, Lei Luo, Andrea Vedaldi, Christian Richardt, Andrea Tagliasacchi, Bo Zhu, and Numair Khan. Free-range Gaussians: Non-grid-aligned generative 3D Gaussian reconstruction. In *ECCV*, 2026.

Qiuhong Shen, Xingyi Yang, and Xinchao Wang. FlashSplat: 2D to 3D Gaussian splatting segmentation solved optimally. In *ECCV*, pp. 456–472, 2024.

Tianchang Shen, Sherwin Bahmani, Kai He, Sangeetha Grama Srinivasan, Tianshi Cao, Jiawei Ren, Ruilong Li, Zian Wang, Nicholas Sharp, Zan Gojcic, Sanja Fidler, Jiahui Huang, Huan Ling, Jun Gao, and Xuanchi Ren. Lyra 2.0: Explorable generative 3D worlds. In *SIGGRAPH Asia*, 2026.

Yu Sheng, Jiajun Deng, Xinran Zhang, Yu Zhang, Bei Hua, Yanyong Zhang, and Jianmin Ji. SpatialSplat: Efficient semantic 3D from sparse unposed images. In *ICCV*, pp. 26404–26414, 2025.

Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kontschieder. Panoptic lifting for 3D scene understanding with neural fields. In *CVPR*, pp. 9043–9052, 2023.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo Open Dataset. In *CVPR*, pp. 2446–2454, 2020.

Xiangyu Sun, Haoyi Jiang, Liu Liu, Seungtae Nam, Gyeongjin Kang, Xinjie Wang, Wei Sui, Zhizhong Su, Wenyu Liu, Xinggang Wang, and Eunbyung Park. Uni3R: Unified 3D reconstruction and semantic understanding via generalizable Gaussian splatting from unposed multi-view images. In *CVPR*, pp. 33280–33290, 2026.

Stanislaw Szymanowicz, Jason Y. Zhang, Pratul [P.] Srinivasan, Ruiqi Gao, Arthur Brussee, Aleksander Hołyński, Ricardo Martin-Brualla, Jonathan T. Barron, and Philipp Henzler. Bolt3D: Generating 3D scenes in seconds. In *ICCV*, pp. 24846–24857, 2025.

Ayca Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-vocabulary 3D instance segmentation. In *NeurIPS*, 2023.

Lei Tian, Xiaomin Li, Liqian Ma, Hao Yin, Zirui Zheng, Hefei Huang, Taiqing Li, Huchuan Lu, and Xu Jia. CCL-LGS: Contrastive codebook learning for 3D language Gaussian splatting. In *ICCV*, pp. 9855–9864, 2025.

Qijian Tian, Zimeng Wu, Xuhong Wang, Lizhuang Ma, and Xin Tan. GroupForward: Building referable 3D scenes via instance-grouped feed-forward Gaussian splatting. *arXiv:2608.17535 [cs.CV]*, 2026a.

Zefan Tian, Yuteng Ye, Yiheng Zhang, Yuhang Yang, Xueqiang Lv, Shizhou Zhang, Le Liu, and Di Xu. SceneReGen: Generative reconstruction of 3D scenes from a single image. *arXiv:2608.23930 [cs.CV]*, 2026b.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pp. 5998–6008, 2017.

Hanyang Wang, Fangfu Liu, Jiawei Chi, and Yueqi Duan. VideoScene: Distilling video diffusion model to generate 3D scenes in one step. In *CVPR*, 2025a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *CVPR*, pp. 5294–5306, 2025b.

Pengfei Wang, Liyi Chen, Zhiyuan Ma, Yanjun Guo, Guowen Zhang, and Lei Zhang. One2Scene: Geometric consistent explorable 3D scene generation from a single image. In *ICLR*, 2026.

Junyi Wu, Van Nguyen Nguyen, Benjamin Planche, Jiachen Tao, Changchang Sun, Zhongpai Gao, Zhenghao Zhao, Anwesa Choudhuri, Gengyu Zhang, Meng Zheng, Feiran Wang, Terrence Chen, Yan Yan, and Ziyan Wu. Consistent instance field for dynamic scene understanding. In *CVPR*, pp. 3783–3793, 2026.

Xianfeng Wu, Yajing Bai, Minghan Li, Xianzu Wu, Xueqi Zhao, Zhongyuan Lai, Wenyu Liu, and Xinggang Wang. 4DLangVGGT: 4D language-visual geometry grounded transformer. *arXiv:2512.05060 [cs.CV]*, 2025.

Yanmin Wu, Jiarui Meng, Haijie Li, Chenming Wu, Yahao Shi, Xinhua Cheng, Chen Zhao, Haocheng Feng, Errui Ding, Jingdong Wang, and Jian Zhang. OpenGaussian: Towards point-level 3D Gaussian-based open vocabulary understanding. In *NeurIPS*, 2024.

Yuxuan Xie, Xuan Yu, Changjian Jiang, Sitong Mao, Shunbo Zhou, Rui Fan, Rong Xiong, and Yue Wang. PanopticSplatting: End-to-end panoptic Gaussian splatting. In *IROS*, pp. 4067–4074, 2025.

Butian Xiong, Rong Liu, Kenneth Xu, Meida Chen, and Andrew Feng. Splat feature solver. In *ICLR*, 2026.

Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4DGT: Learning a 4D Gaussian transformer using real-world monocular videos. In *NeurIPS*, 2025.

Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, pp. 5188–5197, 2019.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4D Gaussian splatting. In *ICLR*, 2024a.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3D Gaussians for high-fidelity monocular dynamic scene reconstruction. In *CVPR*, pp. 20331–20341, 2024b.

Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3D scenes. In *ECCV*, pp. 162–179, 2024.

Yifan Yin, Zehao Wen, Suyu Ye, Jieneng Chen, Zehan Zheng, Nanru Dai, Haojun Shi, Aydan Huang, Zheyuan Zhang, Alan Yuille, Jianwen Xie, Ayush Tewari, and Tianmin Shu. 3D-Belief: Embodied belief inference via generative 3D world modeling. *arXiv:2605.11367 [cs.CV]*, 2026.

Mijin Yoo, In Cho, Subin Jeon, Jiwoo Lee, Eunbyung Park, and Seon Joo Kim. Scenes as objects, not primitives: Instance-structured 3D tokenization from unposed views. *arXiv:2606.29513 [cs.CV]*, 2026.

Yongsheng Yu, Ziyun Zeng, Haitian Zheng, and Jiebo Luo. OmniPaint: Mastering object-oriented editing via disentangled insertion-removal inpainting. In *ICCV*, pp. 17324–17334, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, pp. 11975–11986, 2023.

Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. GS-LRM: Large reconstruction model for 3D Gaussian splatting. In *ECCV*, pp. 1–19, 2024.

Renhe Zhang, Yuyang Tan, Jingyu Gong, Zhizhong Zhang, Lizhuang Ma, Yuan Xie, and Xin Tan. S2GS: Streaming semantic Gaussian splatting for online scene understanding and reconstruction. In *ICML*, 2026.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pp. 586–595, 2018.

Shuaifeng Zhi, Tristan Laidlow, Stefan Leutenegger, and Andrew J. Davison. In-place scene labelling and understanding with implicit scene representation. In *ICCV*, pp. 15838–15847, 2021.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *Int. J. Comput. Vis.*, 127(3):302–321, 2019.

Shijie Zhou, Hui Ren, Yijia Weng, Shuwang Zhang, Zhen Wang, Dejia Xu, Zhiwen Fan, Suya You, Zhangyang Wang, Leonidas [J.] Guibas, and Achuta Kadambi. Feature4X: Bridging any monocular video to 4D agentic AI with versatile Gaussian feature fields. In *CVPR*, pp. 14179–14190, 2025.

Runsong Zhu, Shi Qiu, Zhengzhe Liu, Ka-Hei Hui, Qianyi Wu, Pheng-Ann Heng, and Chi-Wing Fu. Rethinking end-to-end 2D to 3D scene segmentation in Gaussian splatting. In *CVPR*, pp. 3656–3665, 2025.

Runsong Zhu, Jiaxin Guo, Xiaoyang Guo, Zhengzhe Liu, Ka-Hei Hui, Wei Yin, Kai Chen, Wei Chen, Weiqiang Ren, Yunhui Liu, Pheng-Ann Heng, and Chi-Wing Fu. EPS3D: End-to-end feed-forward 3D panoptic segmentation. In *ICML*, 2026.

Zhengyu Zou, Hao Li, Kuixuan Jiao, Liu Liu, Tingyang Xiao, Xiaolin Zhou, Fangzhou Hong, Zhizhong Su, Dingwen Zhang, and Ziwei Liu. IGGT4D: Streaming 4D instance-grounded geometry transformer. *arXiv:2607.19228 [cs.CV]*, 2026.

APPENDIX

The appendix covers implementation details (Appendix A), additional experiments (Appendices B and C), application experiments (Appendix D), and datasets (Appendix E).

A IMPLEMENTATION DETAILS

We report the hyperparameters and implementation choices that determine teacher-target construction, projector pretraining, LucidLyra training, and inference. We will release our code, preprocessed datasets and benchmark annotations upon acceptance.

A.1 TEACHER TARGETS AND FEATURE COMPRESSION

Instance supervision. We use SAM1 ViT-H (Kirillov et al., 2023) to initialize SAM2.1 Hiera-Base+ (Ravi et al., 2025) on the shared conditioning frame 0, followed by independent propagation through each 121-frame trajectory. We use the SAM automatic mask generator with 32 points per side, predicted-IoU and stability thresholds of 0.84 and 0.86, one crop layer, a crop-grid downscaling factor of two, box and crop NMS thresholds of 0.7, and a minimum mask area of 100 pixels. We do not threshold SAM2 logits and merge redundant or nested tracks at a mask-coverage threshold of 0.5. Part-level targets retain SAM multimask candidates 0 and 1 and use the same propagation and consolidation procedure.

Semantic supervision. We build upon RADSeg (Alama et al., 2026) and C-RADIOv3-B (Ranzinger et al., 2024) with its 1152-dimensional SigLIP2 (Tschannen et al., 2025) adapter, a crop size of 336, stride 224, and SCRA and SCGA scales of 10. RADSeg class prediction and SAM refinement are not used. We project the dense tokens to 128 dimensions, normalize them, and resize them bilinearly. We pool all pixels and frames belonging to a track, which yields one pixel-weighted semantic centroid per track and trajectory. This is performed in the object-level instance tracks as well as the part-level tracks.

Constructing semantic and instance targets for the full corpus required an aggregate of 1580 NVIDIA A100 80 GB GPU-days offline. This is a one-time training-data preprocessing cost.

Vision-language feature dimensionality reduction. P_{VL} sums a $1152{\to}128$ linear projection and a $1152{\to}4608{\to}128$ two-layer ReLU MLP. Image and text outputs share the same projection and are ℓ_2-normalized afterward. We train the projector with frozen SigLIP2 SO400M/16 NaFlex features and the SigLIP alignment objective on filtered LAION-2B (Schuhmann et al., 2022). During training, we use a global batch size of 16,384 across eight GPUs, AdamW optimizer with learning rate 10^{-3}, $\beta = (0.9, 0.999)$, $\epsilon = 10^{-8}$, matrix-only weight decay 0.2, 10k warm-up steps followed by cosine decay, and gradient clipping at 1. The SigLIP2 logit bias remains frozen, its scale is capped at 100, and images use the native processor without additional augmentation. We train for 32 epochs and keep the projection network frozen afterward. Training takes 2 days using 8 NVIDIA A100 80 GB GPUs.

A.2 LUCIDLYRA TRAINING

We use eight shared layers in the decoder, followed by eight separate layers for reconstruction and understanding each. We initialize the C-RADIO injection gate to 0.05, and its learnable latent-time weights as $w_\ell = (1 - \ell/15)^2$. We predict raw Gaussian parameters using the reconstruction head. Each static Gaussian additionally carries a 128-dimensional semantic feature, a 12-dimensional object feature, and a 4-dimensional part feature. We clamp semantic and instance outputs to $[-10, 10]$, with object and part channels normalized separately. We optimize all decoder parameters at 704×1280 resolution using AdamW with $\beta = (0.9, 0.999)$, $\epsilon = 10^{-8}$, weight decay 10^{-2}, and gradient clipping at 1. We train on eight GPUs, processing one scene each, without gradient accumulation. We set the learning rates to 10^{-5} for the reconstruction backbone and 10^{-4} for the understanding branch, heads, and conditioning modules. We keep the learning rate constant after 100 warm-up steps. Table 3 summarizes the training curriculum.

Here, L is the number of source frames per trajectory, V is the number of input trajectories, S is the number of supervision views per scene, and B is the per-GPU batch size. The corresponding

Table 3: **LucidLyra training setup.**

Stage	$H \times W$	L	V	S	B	Steps
Static						
1	704×1280	121	1	9	1	10k
2	704×1280	121	1	9	1	20k
3	704×1280	121	1	9	1	20k
4	704×1280	121	1–6	$3V$	1	10k
Dynamic						
5	704×1280	121	1–6	12	1	10k

global batch size is eight. Stages 1–4 respectively denote auxiliary initialization, single-trajectory training with object targets, single-trajectory training with part targets, and multi-trajectory training with part targets. Opacity pruning is applied during the last 1000 steps of Stage 4. Stage 5 initializes from stage 4 checkpoint before opacity pruning. For Stage 5, $S = 12$ for reconstruction, depth, and semantic supervision, whereas the instance loss uses a random six-trajectory (forward or backward) block at a time ($S = 6$). The opacity pruning loss is applied for the last 1000 steps.

We set the weights of the RGB, LPIPS, and depth losses to 1, 0.5, and 0.05. For the rendered and direct per-Gaussian semantic losses, we use weight 0.5 during stage 1 and 4 thereafter. We set the weights of the object and part contrastive losses to 0.05. For the static model, from multi-trajectory training onward, we restrict the reconstruction to valid object-track pixels, and gradients from rendered feature losses also update Gaussian geometry. We use the reconstruction mask to exclude untracked background via ignored SAM pixels. During the pruning stage, we retain the highest-opacity 20% of Gaussians for rendering and direct semantic supervision, and add an opacity regularizer equal to 0.1 times their mean activated opacity. For object and part losses independently, we sample at most 4,096 tracked pixels per scene across the target views, using replacement when subsampling, and use a temperature of 0.05. Anchors must have at least one positive, and part anchors must additionally have a negative from another part of the same object. We use $\epsilon = 10^{-8}$ for feature normalization, and 10^{-6} for logarithms.

For dynamic training, we zero-initialize the time-conditioning weights and sample one to six original source trajectories and a query time uniformly. All 12 original and time-reversed target trajectories provide reconstruction, depth, and semantic supervision. The instance loss uses one randomly selected six-trajectory block with consistent instance IDs (Section 3.5). Since the preprocessing of the reversed trajectories is seeded with the last frame of the original video, the instance IDs are different from those of the forward trajectories. C-RADIO conditioning uses the queried-time frame, and reconstruction uses valid object-track support. Temporal prototype matching uses a cosine-similarity threshold of 0.2. Training the static and dynamic Gaussian decoders takes 10 days using 8 NVIDIA A100 80 GB GPUs.

A.3 Inference and Evaluation Protocol

Inference. We use batch size one and all six input trajectories. At inference, we prune the lowest-opacity 80% of Gaussians, retaining 20%. LucidLyra+ uses the same checkpoint and retained Gaussians, changing only semantic readout and the resulting instance grouping. We ℓ_2-normalize Gaussian features and 128-D text embeddings obtained by projecting the SigLIP2 features from 1152 dimensions. The learned instance embeddings are class-agnostic, but inference clusters them within predicted semantic classes. For each class, we select the $\lceil 0.5N \rceil$ highest-opacity Gaussians, and average each normalized object embedding with $k = 20$. For all datasets, object clustering uses Euclidean HDBSCAN with minimum cluster size 1000, minimum samples 5, $\epsilon = 0.06$, leaf selection, and a single cluster allowed. Unassigned Gaussians are assigned to the nearest neighbor. For part-level instance grouping, we cluster the final four instance channels using an opacity threshold of 0.1, Euclidean HDBSCAN, minimum cluster size 100, minimum samples 5, $\epsilon = 0.01$, EOM selection, and allow for a single cluster. Static LucidLyra+ uses the RADSeg+ prediction of the input image; dynamic LucidLyra+ applies RADSeg+ anchoring to the input frame at each queried time. Labels are nearest-neighbor resized to the source-ray grid and gathered using the retained ray indices. For the prototype assignments, we use cosine similarity ≥ 0.95. For dynamic inference,

we decode and cluster each queried state independently using the same object-clustering settings, a top-opacity ratio of 0.5, assignment $k = 20$, and a temporal similarity threshold of 0.2.

Metrics and Comparison Protocol

We perform the static evaluation of all trajectories at frames $\{0, 20, 40, 60, 80, 100, 120\}$, and the dynamic evaluation of all six trajectories at queried times $\{0, 40, 80, 120\}$. We compute PSNR on RGB values clamped to $[0, 1]$, SSIM with an 11×11 Gaussian window, and VGG-LPIPS with normalized inputs. Semantic evaluation reports mIoU and mAcc using the ADE20K classes present in each reference scene. Instance evaluation clips predictions and references to annotated thing regions. These reference-conditioned scores measure classification within a known vocabulary and grouping within known foreground.

We evaluate instance consistency across views. We follow video instance segmentation (Yang et al., 2019) in representing identities as tubes and computing tube IoU as the ratio of summed intersections to summed unions across observations within each scene. At each threshold $\tau \in \mathcal{T} = \{0.25, 0.30, \ldots, 0.75\}$, Hungarian assignment maximizes the number of one-to-one matches with tube IoU $\geq \tau$. Matched pairs, unmatched predictions, and unmatched references define TP, FP, and FN, respectively. We pool these counts across scenes and define the track precision–recall product:

$$\mathrm{TrackPR}_\tau = \frac{\mathrm{TP}_\tau}{\mathrm{TP}_\tau + \mathrm{FP}_\tau} \frac{\mathrm{TP}_\tau}{\mathrm{TP}_\tau + \mathrm{FN}_\tau}, \qquad \mathrm{mTrackPR} = \frac{1}{11} \sum_{\tau \in \mathcal{T}} \mathrm{TrackPR}_\tau. \tag{8}$$

We report mTrackPR and $\mathrm{TrackPR}_{25}$ as percentages, defining zero-denominator ratios as zero. Our clustering produces masks without confidence scores. TrackPR therefore uses an unranked precision–recall product over tube-IoU matches. In contrast, IGGT (Li et al., 2026c) averages tracking mIoU and tracking success rate over reference objects, without directly penalizing unmatched predicted instances. TrackPR counts unmatched predicted tubes as false positives, penalizing oversegmentation within the evaluated thing regions.

Baseline implementations. LangSplatV2, SFS, Unified-Lift, and Chorus use a frozen Lyra (Bahmani et al., 2026) reconstruction. Our lifted-teacher reference changes only the feature fields, assigning our compressed SigLIP2 semantic features and deterministic 12-D Hamming codes to RGB-derived SAM tracks. LangSplatV2 (Li et al., 2025b) independently optimizes three language levels for 10k steps using a 64-entry VQ codebook. SFS (Xiong et al., 2026) lifts SAM ViT-H/OpenCLIP ViT-B/16 features with Tikhonov coefficient 1.2 and temperature 100. Unified-Lift (Zhu et al., 2025) trains a 16-D object field and 256-entry codebook for 30k steps with Adam learning rates 2.5×10^{-3} and 5×10^{-4}; and Chorus (Li et al., 2026g) uses its released language–DINO model. IGGT (Li et al., 2026c) reconstructs geometry from 11 calibrated 518×280 views and applies k=20 feature smoothing followed by HDBSCAN (ϵ=0.01, minPts=100, MCS=500). Using more input views results in OOM errors and significantly increased inference time due to the HDBSCAN clustering while not improving the instance segmentation metrics. For dynamic baselines, 4D LangSplat (Li et al., 2025c) fits two 100-epoch autoencoders and four 24k-step fields to all 726 observations; TRASE (Li et al., 2026h) trains a 32-D field for 30k steps. IGGT4D (Zou et al., 2026) streams all views through its fixed ViT-G/8-D model in rewrite mode, using $0.90/0.85$ match/grow thresholds and LSeg semantics.

B Additional Results

B.1 Dynamic Reconstruction and Understanding

Next, we demonstrate that our method can be extended to dynamic 3D scene generation from a monocular input video. Table 4 compares LucidLyra queried-time scene states with dynamic reconstruction and understanding baselines in the LucidLyra-SDG Dynamic evaluation dataset.

Semantic and instance understanding. LucidLyra achieves 53.2 mIoU and 69.3 mAcc; LucidLyra+ increases these to 62.6 and 79.3, respectively. Their mTrackPR scores are 11.7 and 12.3, with $\mathrm{TrackPR}_{25}$ scores of 19.7 and 20.4.

Reconstruction. LucidLyra obtains 21.88 dB PSNR and 0.252 LPIPS, compared with Lyra's 22.16 dB and 0.248. This represents a minimal reconstruction trade-off for dynamic scene understanding.

Table 4: **Dynamic reconstruction and understanding on LucidLyra-SDG Dynamic.** We report PSNR (dB), LPIPS (–), and mIoU, mAcc, mTrackPR, and TrackPR$_{25}$ (all in %). † denotes using the video diffusion model's RGB output.

Method	*Reconstruction*		*Semantics*		*Instance*	
	PSNR↑	LPIPS↓	mIoU↑	mAcc↑	mTrackPR↑	TrackPR$_{25}$↑
Lyra + Lifted Teachers	22.16	**0.248**	53.2	69.4	12.1	15.1
Lyra	22.16	**0.248**	–	–	–	–
Lyra† + 4D LangSplat	**23.03**	0.354	30.3	47.8	–	–
Lyra† + TRASE	22.77	0.309	–	–	7.7	12.5
Lyra† + IGGT 4D	–	–	36.1	46.0	7.8	12.9
LucidLyra *(Ours)*	21.88	0.252	53.2	69.3	11.7	19.7
LucidLyra+ *(Ours)*	21.88	0.252	**62.6**	**79.3**	**12.3**	**20.4**

Our dynamic extension represents an initial step toward structured 4D scene generation and understanding, with encouraging results that motivate further development in this direction.

B.2 Compact Vision-Language Representation

Since the semantic feature dimension directly affects the cost of representing each Gaussian, we train a projection network inferring a lower-dimensional compressed version of the SigLIP2 latent space. Next, we evaluate the projector independently of LucidLyra on two tasks: zero-shot ImageNet-1K classification using SigLIP2 SO400M/16 NaFlex, and RADSeg segmentation on the 2,000-image ADE20K validation set. In both tasks, we project image and text descriptors into the same compact space. The segmentation analysis uses three inference resolutions following Alama et al. (2026).

Table 5: P_{VL} **projection quality.** SigLIP2 ImageNet val zero-shot classification accuracy and RADSeg open-vocabulary semantic segmentation on ADE20K eval mIoU. Both metrics in %↑.

Method	Feature Dim.	*ImageNet*		*RADSeg on ADE20K*		
		Top-1↑	Top-5↑	Low↑	Mid↑	High↑
SigLIP2	1152	83.50	96.86	27.16	28.96	28.94
+ Projection P_{VL}	256	80.69	95.89	25.55	27.06	27.09
+ Projection P_{VL}	128	76.61	94.02	24.22	25.34	25.14
+ Projection P_{VL}	64	70.16	90.43	22.13	23.66	23.61

Table 5 shows the accuracy–capacity trade-off. The 256-dimensional projector reduces descriptor width by $4.5\times$, with a 2.81-point decrease in ImageNet top-1 accuracy and a 1.79-point decrease in ADE20K mIoU averaged over the three resolutions. Our selected 128-dimensional representation reduces width by $9\times$ and achieves 76.61% top-1 accuracy and 24.90 mean mIoU, decreasing 6.89 and 3.45 points, respectively. Although 256 dimensions better preserve the native representation, that configuration exceeds the GPU budget when training on 80GB GPUs. Further reducing the feature dimension to 64 substantially degrades classification and segmentation accuracy. Hence, we use 128 dimensional semantic representations.

C Qualitative Results

We show static and dynamic examples, compare baseline predictions, and examine object- and part-level instance fields. We further provide qualitative video results at https://anonlucidlyra.github.io.

Figure 4 shows qualitative LucidLyra scene generation and understanding across diverse generated examples. The qualitative results in Figs. 4 and 5 show crisp reconstructions and coherent open-vocabulary and instance segmentation across diverse scenes and queried categories, while the feature PCA visualizations suggest clear separation of semantic concepts and object instances in the learned representation. Figure 5 shows dynamic qualitative LucidLyra results from generated inputs. Dynamic LucidLyra is able to reconstruct the motion and provides temporally consistent open-vocabulary and instance segmentation. In Fig. 6 we show qualitative results of LucidLyra generative 3D scene reconstruction and understanding on random Waymo Open (Sun et al., 2020) images. LucidLyra generalizes well to the real-image conditioned driving scenes.

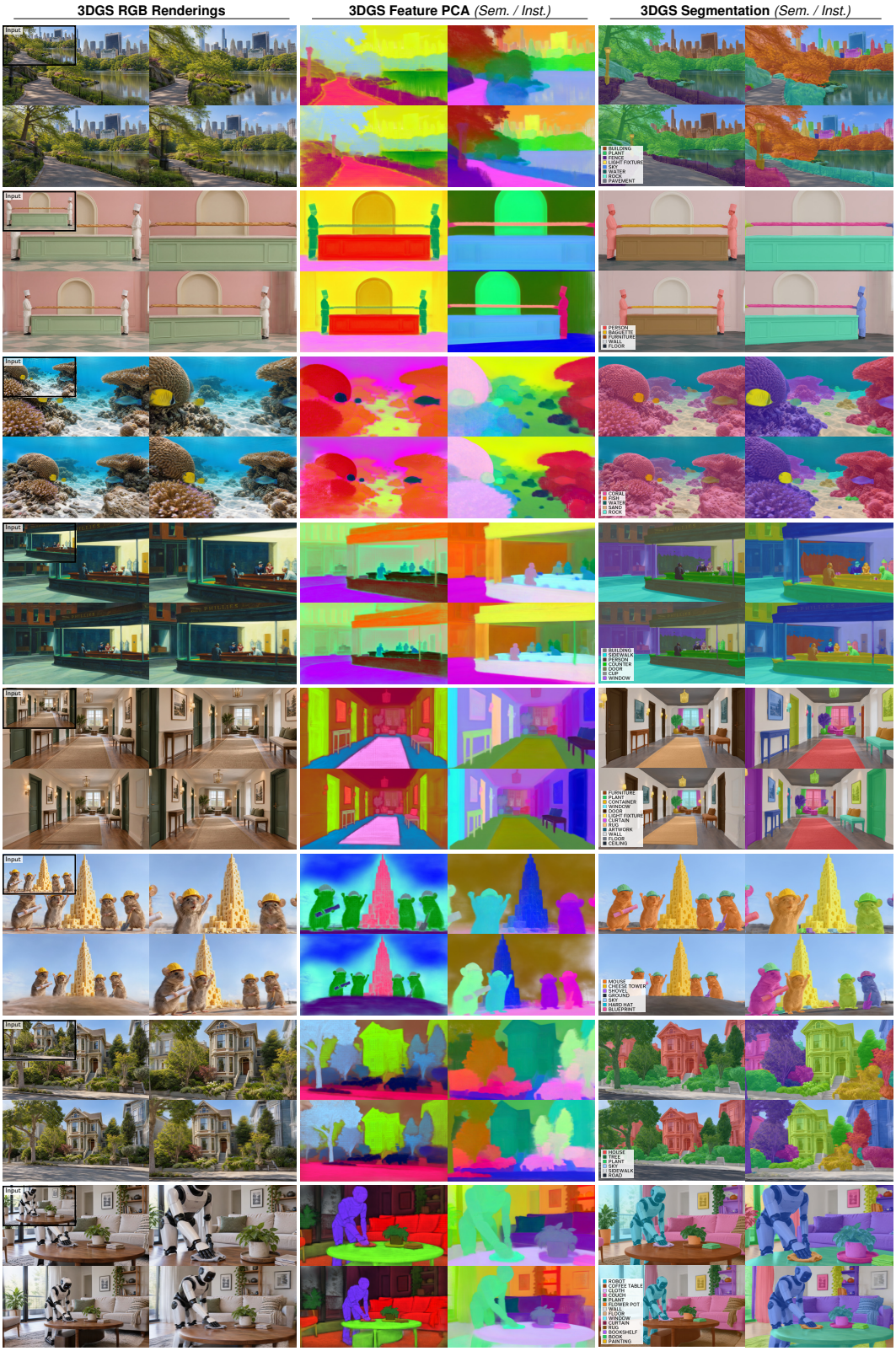


Figure 4: **LucidLyra qualitative results.** We visualize input- and novel-view 3DGS renderings across RGB, semantic/instance feature PCAs, and open-vocabulary/instance segmentation.

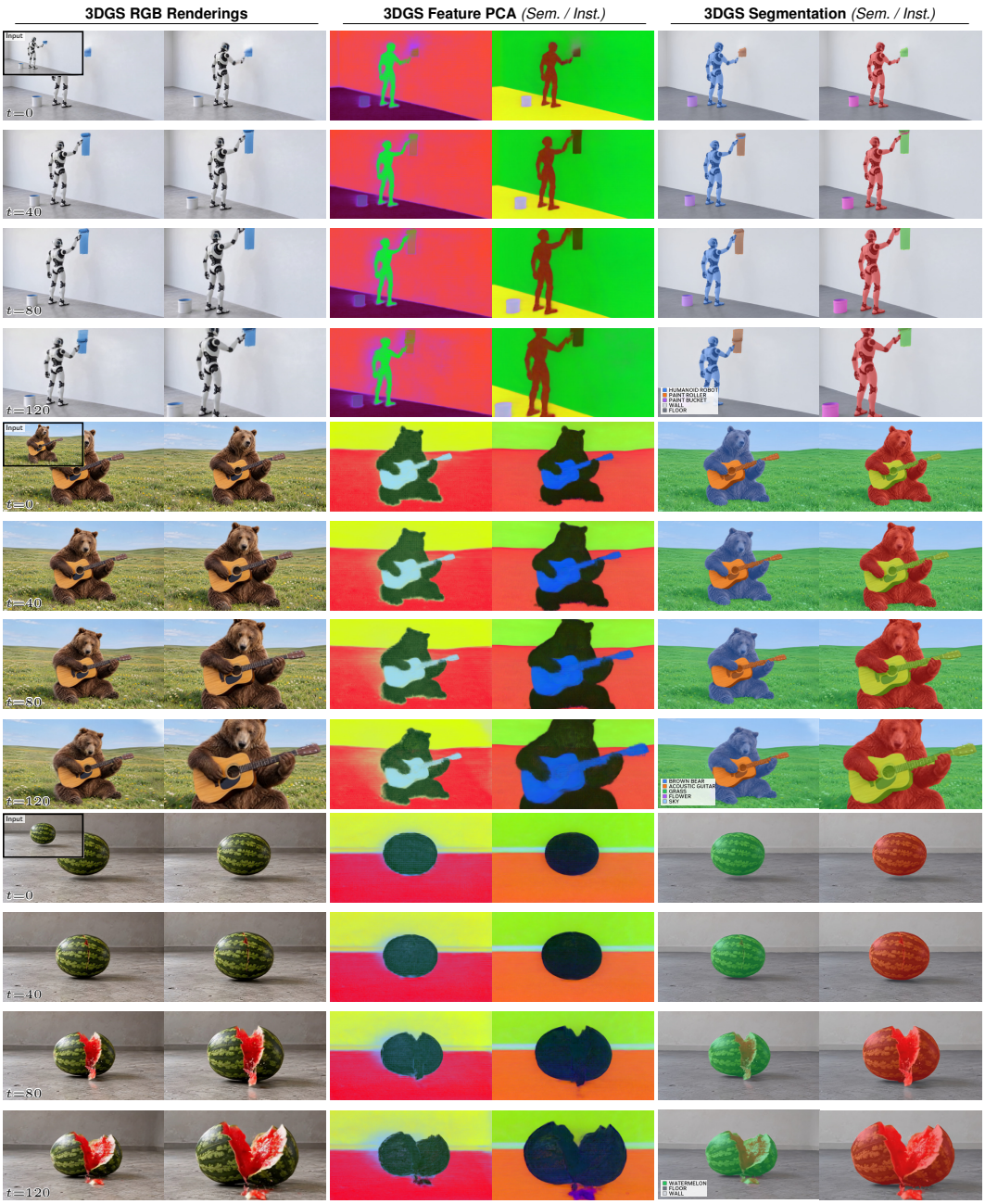


Figure 5: **Dynamic LucidLyra qualitative results.** We visualize input- and novel-view 3DGS renderings across RGB, semantic/instance feature PCAs, and open-vocabulary/instance segmentation.

C.1 Qualitative Baseline Comparison

Figures 7 and 8 compare semantic and instance predictions across both static benchmarks.

Open-vocabulary semantics. LangSplatV2 and SFS recover broad semantic structure but exhibit class confusion and label leakage across object boundaries, with LangSplatV2 additionally showing flickering artifacts as the viewpoint changes. The feed-forward baselines Chorus and IGGT exhibit incomplete category coverage and noisy boundaries, respectively. We hypothesize that their reduced performance partly reflects a domain shift from real-world and synthetic 3D training data to our generatively reconstructed scenes. LucidLyra preserves coherent semantic regions across viewpoints,

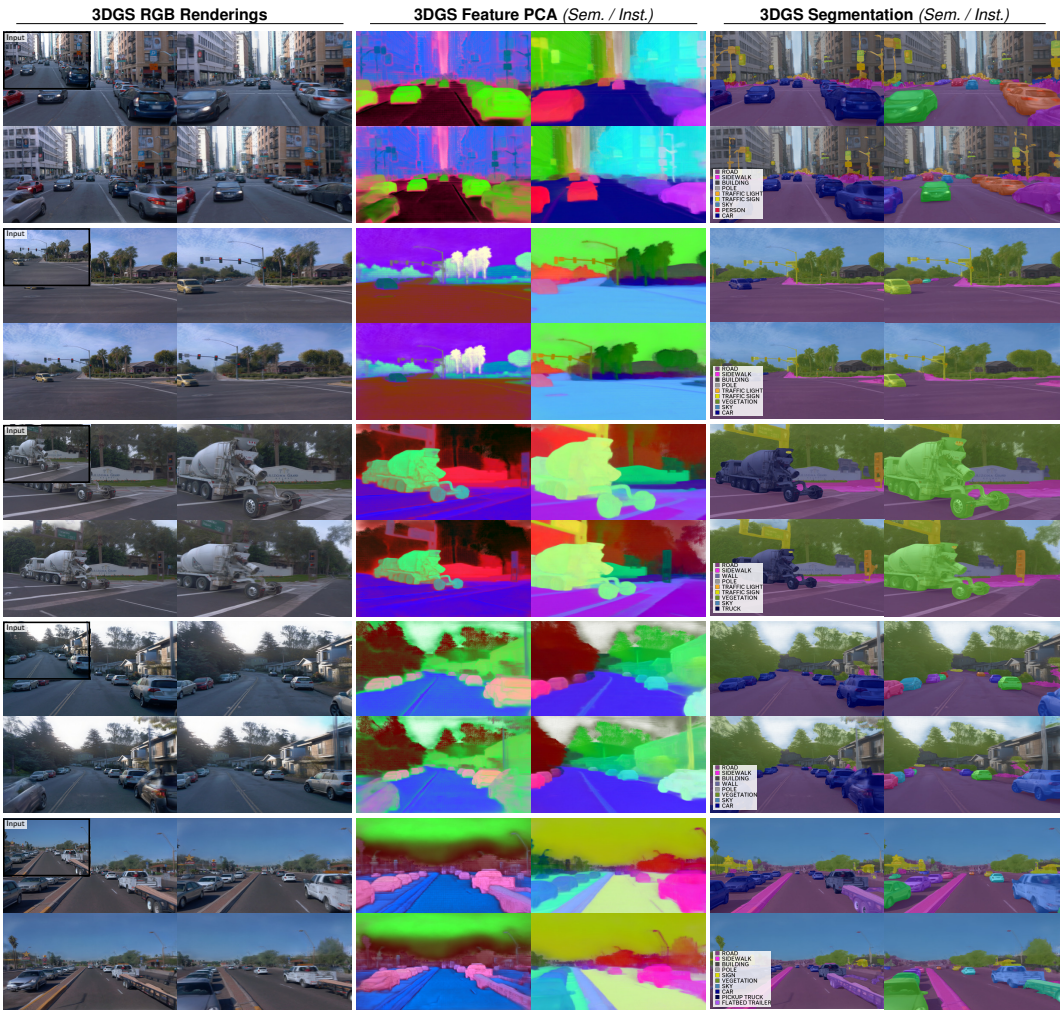


Figure 6: **Waymo qualitative examples.** Each two-row block shows one scene from four viewpoints. Paired columns show RGB, semantic/instance feature PCA, and semantic/instance segmentation. Insets show the original input images.

approaching the lifted-teacher reference while correcting some of its assignment errors. LucidLyra+ further refines semantic assignments without changing geometry or appearance.

Instance segmentation. Unified-Lift produces coherent instance regions but exhibits local label fragmentation and bleeding across object boundaries. IGGT shows blurred instance boundaries and a tendency to over-segment individual objects. Its reconstruction appears sensitive to cross-view inconsistencies in the generated videos, producing duplicated fine-scale structures that further blur segmentation predictions. Our methods preserve more coherent object-level partitions across viewpoints, while the lifted-teacher reference remains competitive. Since instance clustering is conditioned on predicted semantic classes, LucidLyra+ can modify instance partitions without changing the learned instance embeddings.

Dynamic scenes. In the dynamic setting, LucidLyra better separates foreground objects from background surfaces and preserves finer semantic regions than 4D LangSplat (Li et al., 2025c) and IGGT4D (Zou et al., 2026) (Fig. 9). The dynamic baselines require different scene representations and cannot be directly applied to our independently decoded 3D Gaussian states. We therefore evaluate their respective reconstructions and report reconstruction and understanding metrics separately (Table 4).

For instance segmentation, LucidLyra maintains coherent foreground masks across queried times, with fewer local gaps and cleaner boundaries than IGGT4D. TRASE (Li et al., 2026h) also recovers coherent foreground regions but produces coarser background partitions. The lifted-teacher reference remains competitive, particularly in separating smaller objects, while our method preserves complete foreground regions with less fragmentation.

C.2 Hierarchical Instance Segmentation

LucidLyra infers two levels of instance representations that we aim to analyze qualitatively in Fig. 10. The main object-level instance features group scene entities, while additional part-level features expose finer regions within the recovered objects. The qualitative results reveal a consistent feature hierarchy across all examples: object-level instance features remain similar over the spatial extent of an object, enabling clustering to recover a single instance mask, whereas part-level features preserve finer variations that distinguish individual object components. Because these representations form continuous feature fields, occasional grouping errors may still occur. While open-vocabulary querying and object-level instance segmentation provide the foundation for interacting with our generated scenes, part-level features offer a lightweight extension that enables finer-grained control over individual object components.

D Downstream Applications

LucidLyra generates structured, queryable scenes that support diverse downstream applications. In the following section, we demonstrate this capability through examples of robotic simulation, 3D scene editing, and 3D asset mining. See https://anonlucidlyra.github.io for video demos.

D.1 Robot Simulation

We extract a mesh from the LucidLyra depth maps and camera parameters by computing a signed distance field on a hierarchy of sparse grids. We use the generated scene, extracted mesh and the semantic labels to set up a simulation in NVIDIA IsaacSim (NVIDIA, 2026a). For navigation, the semantic labels partition the scene into free space and obstacles, over which we build an occupancy grid and plan a route to a semantically specified goal using a clearance-weighted Dijkstra search. A robot (Unitree G1 humanoid) then follows this route with a pure-pursuit controller while a learned rough-terrain locomotion policy drives its joints under full rigid-body physics. It also carries a box rigidly attached near its torso to demonstrate loco-manipulation. For manipulation, the LucidLyra semantic labeling localizes a kitchen counter and sink in a separate reconstruction, from which we derive a mount pose and reach-constrained pick-and-place targets for a stationary Franka robotic arm. The arm is simulated with full articulated-body dynamics and a friction-based parallel-jaw grasp under differential inverse-kinematics control, picks a small synthetic object (a sphere) from the sink and places it on the counter. The extracted mesh serves as the collision proxy during

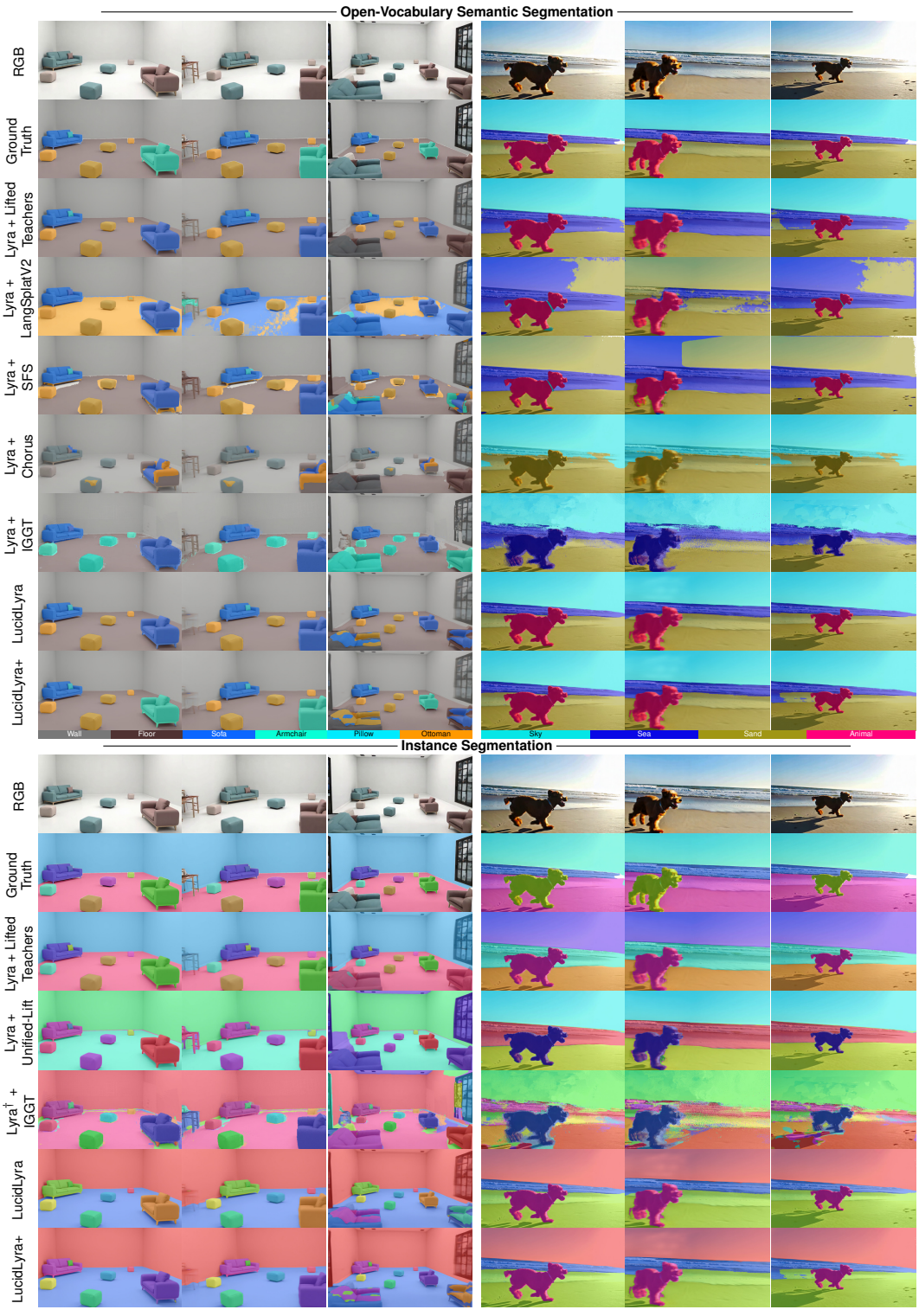


Figure 7: **Qualitative comparison on LucidLyra-SDG Static eval.** We show three viewpoints per scene, with semantic predictions above and instance predictions below. † denotes directly using generated multi-view RGB.



Figure 8: **Qualitative comparison on LucidLyra-ADE20K static eval.** We show three viewpoints per scene, with semantic predictions above and instance predictions below. † denotes directly using generated multi-view RGB.

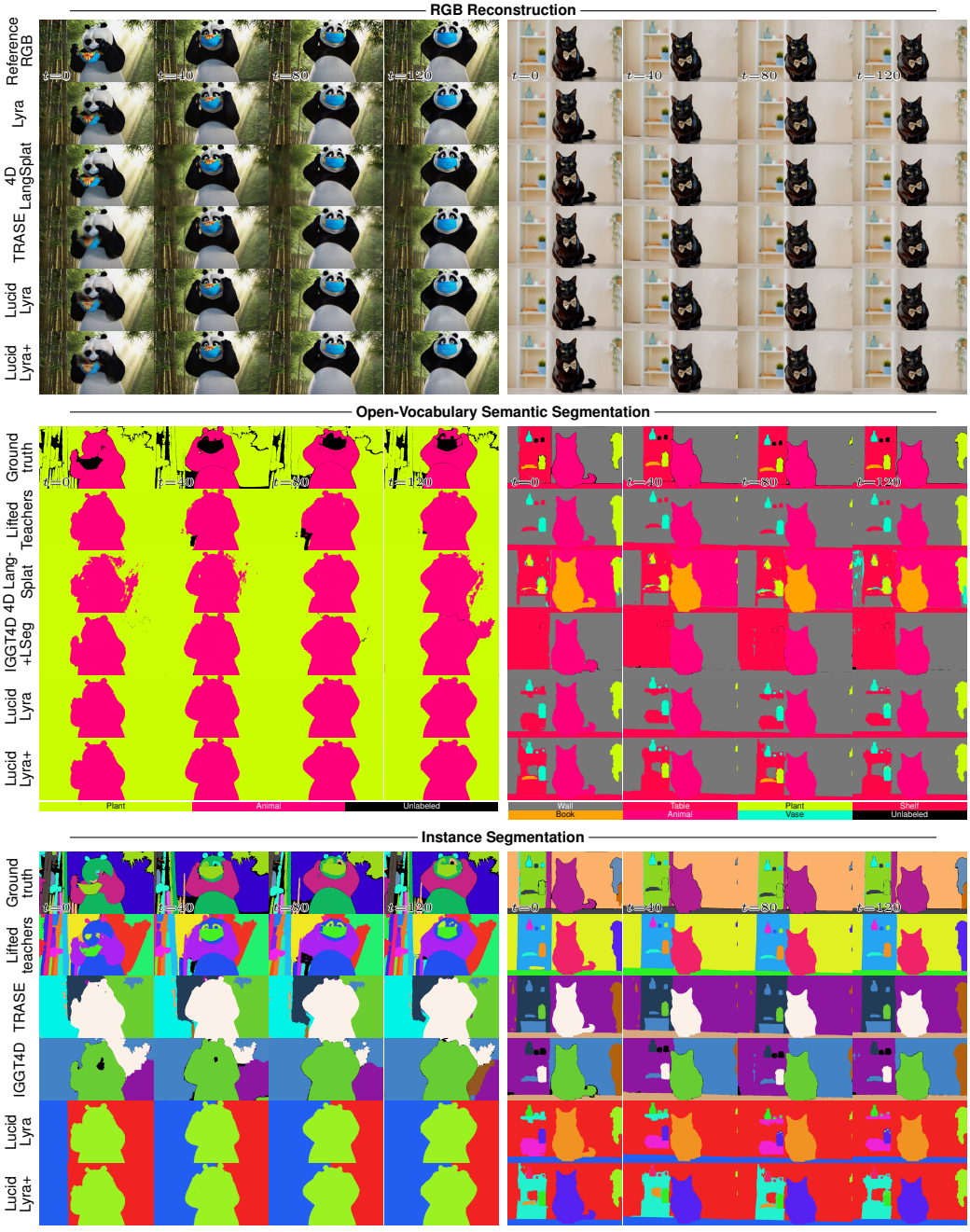


Figure 9: **Qualitative comparison on the dynamic evaluation set.** We show times $t = 0, 40, 80, 120$ along an orbiting camera trajectory. Both scene time and camera pose change. From top to bottom: RGB reconstruction, semantic segmentation, and instance segmentation. Dynamic Lyra supplies the reconstruction used by the lifted-teacher baseline. Instance ID colors are random per method.

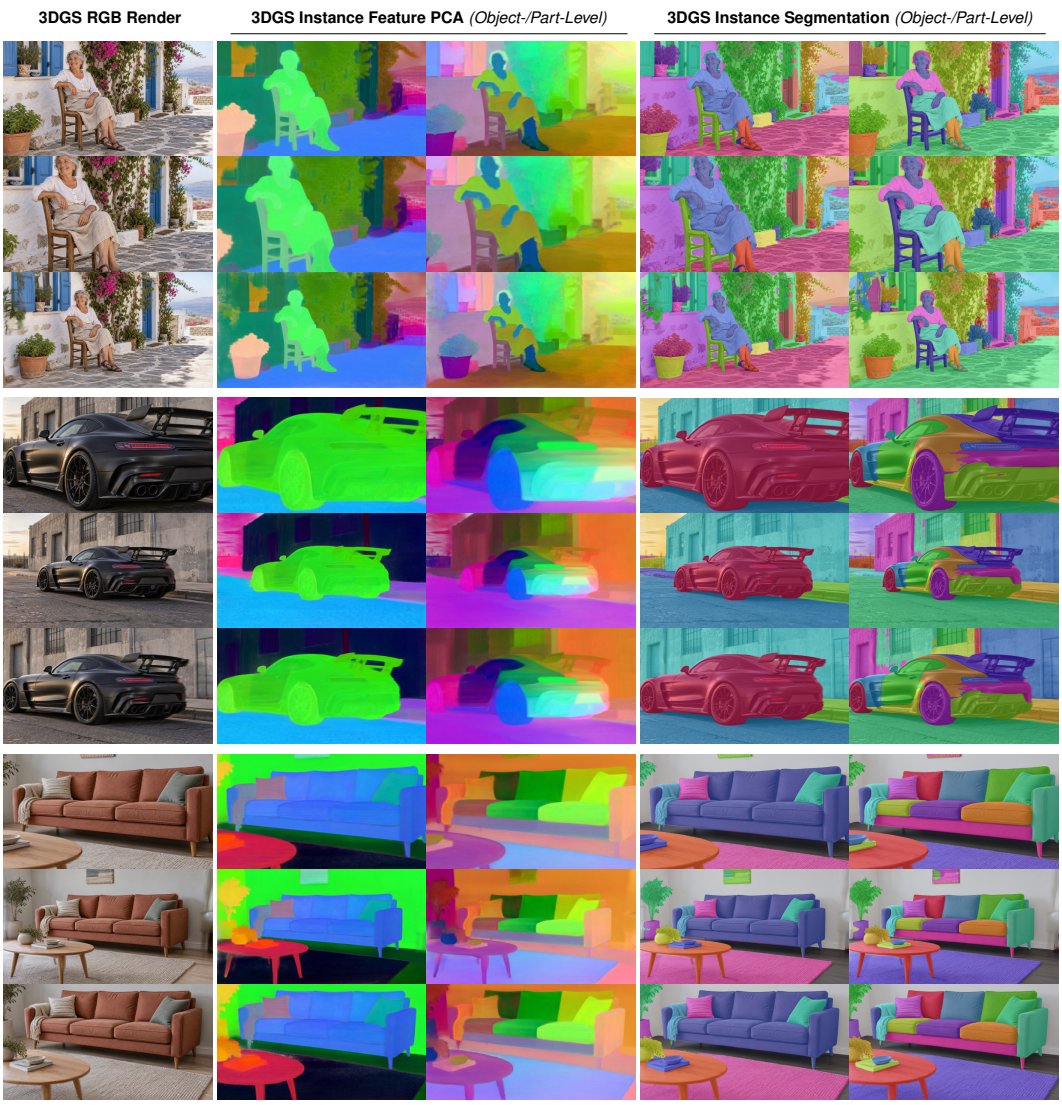


Figure 10: **Object- and part-level instance segmentation.** We visualize 3 views per scene, showing 3DGS RGB render, object- and part-level 3DGS feature PCA, 3DGS object clusters, and within-object part clusters. Random colors denote cluster IDs.

simulation, while the original Gaussian-splat reconstruction is rendered for visual fidelity. Figure 11 visualizes the simulation examples.



Figure 11: **Robot simulation in NVIDIA Isaac Sim using LucidLyra scenes.** Four different robot-simulation scenarios built on LucidLyra reconstructions. Extracted geometry provides collision surfaces, semantic labels support task planning, and Gaussian-splat rendering preserves the scene appearance. Overlaid robot poses show motion over time, with increasing opacity indicating temporal progression.

D.2 3D Scene Editing.

We implement a simple browser-based editor. The user can select objects through text queries or clicks. Text selection intersects the predicted semantic class with an instance ID, thereby separating objects of the same category, while click selection retrieves the semantic–instance pair of the Gaussian center nearest the camera ray. Users can further refine the selection using separate similarity thresholds for Gaussian semantic and instance features relative to prototypes computed from the initial support.

Translation, rotation, scaling, duplication, and deletion operate directly on the selected Gaussian parameters. Transformations are applied around the selection centroid and update positions, anisotropic scales, and orientation quaternions, while color, opacity, and learned features remain attached to each primitive. Duplication appends a transformed copy, whereas deletion removes the selected primitives. Moving or deleting an object can expose unobserved regions. After the user commits all edits, OmniPaint (Yu et al., 2025) inpaints the original object support, and a subsequent LucidLyra forward pass reconstructs a clean background scene. We retain only reconstructed Gaussians near the edit whose projections fall within the exposed source support, excluding the transformed object's footprint. Opacity tapering converts these Gaussians into a smoothly blended local background patch. Figure 12 visualizes our GUI as well as editing examples.

D.3 3D Asset Mining

Much like in 3D scene editing, we can make use of the appearance from multiple views, together with the instance and semantic segmentation masks, to mine high-quality 3D assets from the generated scenes.

For each selected object, we extract six segmented views, prioritizing viewpoint diversity and object coverage without occlusion. The segmented RGB observations together with the corresponding



Figure 12: **LucidLyra 3DGS Scene Editing.** *Left:* Our GUI provides an interface for natural interaction with the generated Gaussian scenes. Gaussians are selected via text prompts or via clicking and can then be manipulated interactively. *Right:* Editing examples showing the initial 3DGS scene, the applied edits and the final scene.

camera parameters are provided to the pretrained Pixal3D (Li et al., 2026b) multi-view reconstruction model to produce a textured mesh. Figure 13a presents assets extracted from LucidLyra scenes. Comparisons in Fig. 13b show that the incorporation of multi-view observations can substantially improve structural consistency compared to single-view reconstruction without the scene generation.

Furthermore, the generated meshes can also be post-processed by mesh-to-articulation pipelines. In Fig. 13c, we employ PARTICULATE (Li et al., 2026d) to estimate moving parts and the corresponding articulation kinematic tree. The quality of the recovered articulation remains limited by the completeness and physical plausibility of the generated asset.

E Datasets and Annotations

We augment the PhysicalAI-SpatialIntelligence-Lyra-SDG (Bahmani et al., 2026) with semantic and instance targets. For evaluation, we introduce three benchmarks based on PhysicalAI-SpatialIntelligence-Lyra-SDG and ADE20K (Zhou et al., 2019).

E.1 LucidLyra-SDG Static

Training data. The static Lyra-SDG corpus comprises 59 031 image-conditioned scenes spanning highly diverse indoor and outdoor environments, people, animals, and various imaginative settings. Each scene provides six generated camera trajectories of 121 frames at 704×1280, together with depth, camera parameters, and video latents. Figure 14 visualizes random training data samples.

Evaluation data. LucidLyra-SDG contains 25 manually curated diverse scenes held out from the training data. To obtain high-quality reference annotations, we use an ensembled version of our SAM/SAM2 instance preprocessing, combined with human review, refinement, and assignment of ADE20K semantic labels. The resulting labels and persistent identities provide references across all six generated trajectories. Figure 15 shows annotated examples.

LucidLyra-ADE20K uses 25 diverse ADE20K validation images (Zhou et al., 2019), selected for scene-category diversity. Gen3C expands each image into six 121-frame videos. We use a SAM2 ensemble to propagate the original semantic and instance ground-truth labels through all camera trajectories, followed by human review and refinement. Figure 16 shows example scenes.

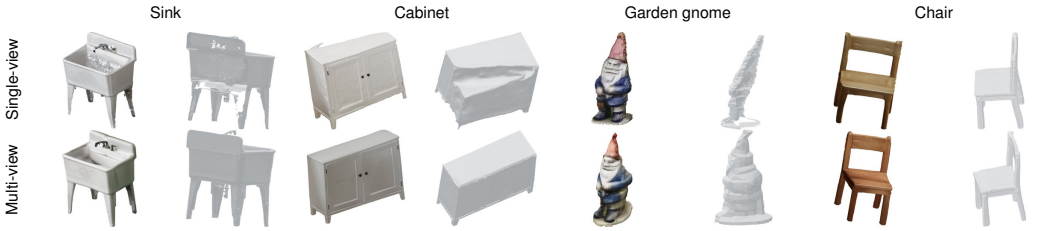
E.2 Dynamic Datasets

Training data. The dynamic Lyra-SDG corpus comprises 7378 video-conditioned scenes. Generation from the original and temporally reversed input video provides two complementary six-trajectory batches, each containing 121 synchronized frames per trajectory. Targets are constructed within each batch, keeping their independently initialized track IDs separate. Figure 17 illustrates training scenes across viewpoints and time.

Evaluation data. LucidLyra-SDG Dynamic contains 25 diverse curated scenes excluded from the training set. Annotation follows the static setup. Figure 18 visualizes annotated examples.



(a) Asset selection and extraction.



(b) Single-view vs. multi-view reconstruction.



(c) Estimated articulation of extracted assets.

Figure 13: **Asset extraction and articulation.** *(a)* Selected instances alongside their textured reconstructions. Display scales are independent. *(b)* Selected single-view (top) and multi-view (bottom) Pixal3D results. Each result pairs a textured view with an alternate neutral-material view. *(c)* Articulated assets shown closed and open, with original textures and part segmentation color. Part boundaries and hinges are obtained through PARTICULATE (Li et al., 2026d) on the generated asset meshes. Doors open to 85° and lids to 75° for illustration, not as measured joint limits.

F Related Work

Below, we discuss further related work not covered in the main paper.

Dynamic representations

Dynamic semantic and instance representations have likewise focused on captured video, spanning per-sequence optimization and feed-forward reconstruction. 4D LangSplat (Li et al., 2025c) fits language features to deformable 4D Gaussians, whereas TRASE (Li et al., 2026h) learns an instance field from per-frame masks. Related methods distill foundation-model features or temporal cues into scene-specific dynamic representations (Labe et al., 2024; Fiebelman et al., 2025; Zhou et al., 2025; Ji et al., 2024; Hu et al., 2025; Wu et al., 2026; Jiang et al., 2026b). Feed-forward approaches amortize language-aligned dynamic reconstruction (Wu et al., 2025; Qiu et al., 2026), while IGGT4D (Zou et al., 2026) jointly predicts streaming geometry and persistent instances.

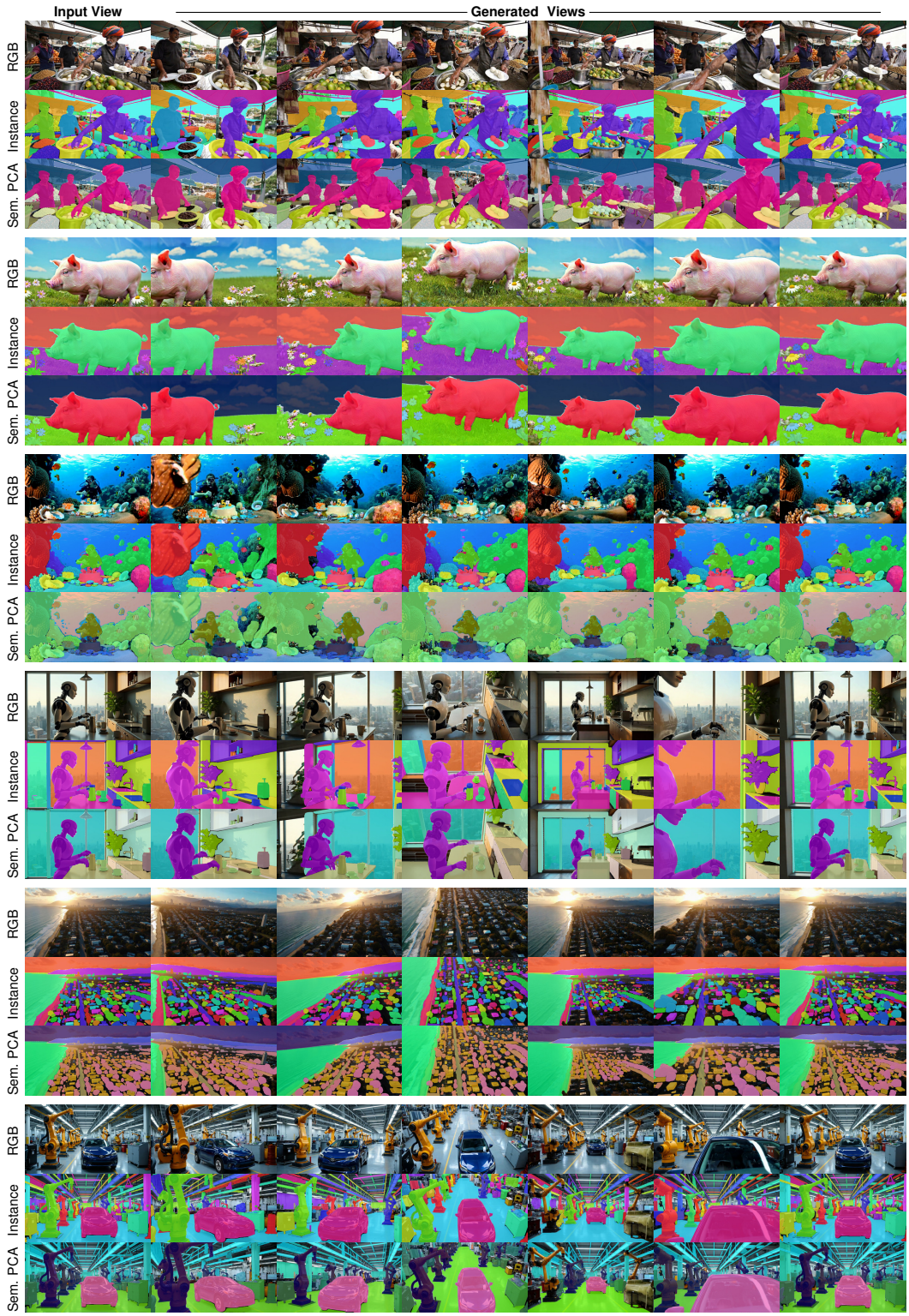


Figure 14: **LucidLyra-SDG static training data.** RGB frames, instance tracks, and PCA visualizations of semantic teacher features across generated camera trajectories.

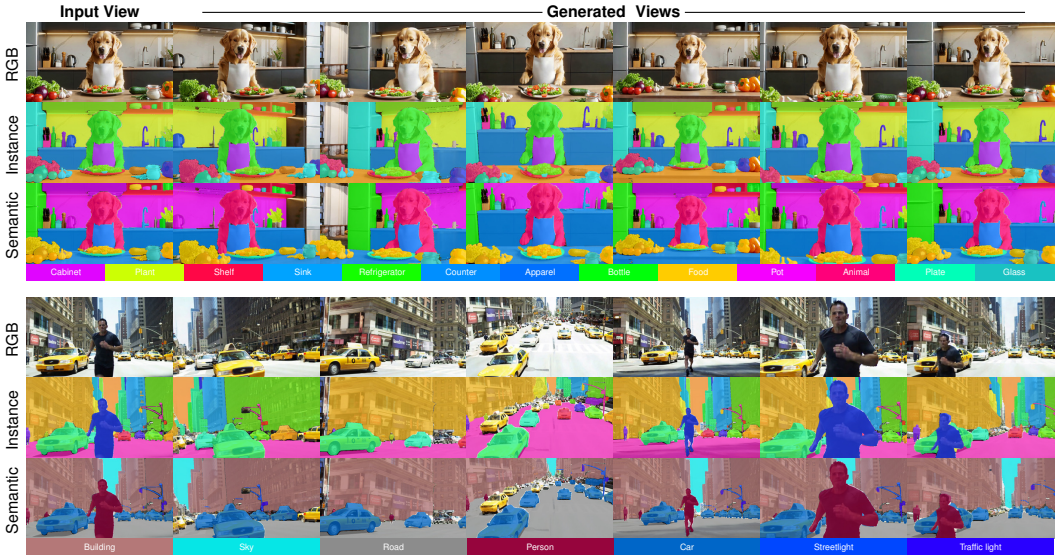


Figure 15: **LucidLyra-SDG static evaluation data.** RGB frames and reference semantic and instance annotations for the conditioning view and generated viewpoints.

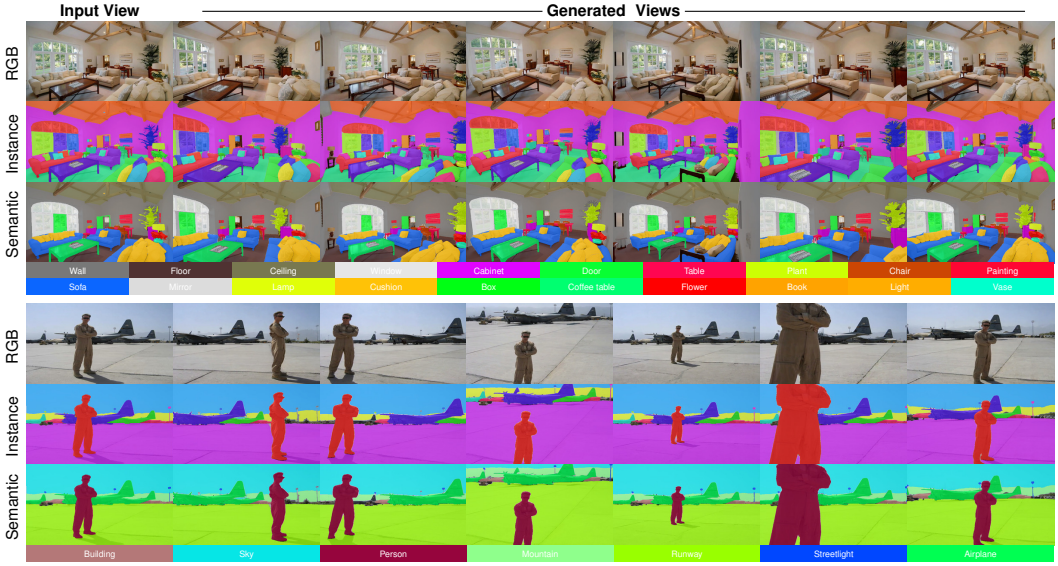


Figure 16: **LucidLyra-ADE20K evaluation data.** Real conditioning images and generated viewpoints with semantic and instance annotations propagated from official ADE20K labels.

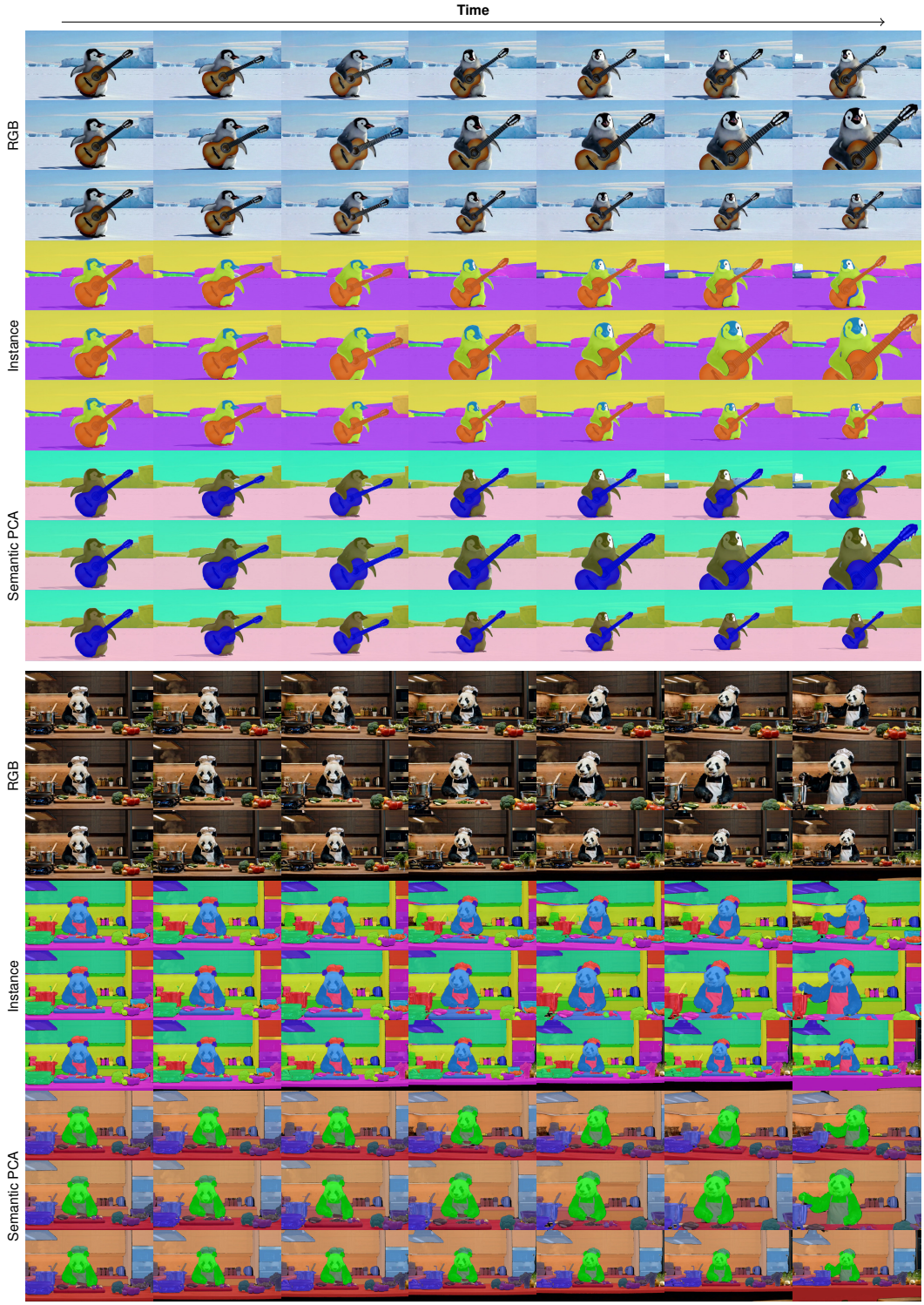


Figure 17: **LucidLyra-SDG dynamic training data.** RGB frames, instance tracks, and PCA visualizations of semantic teacher features across viewpoints and time.

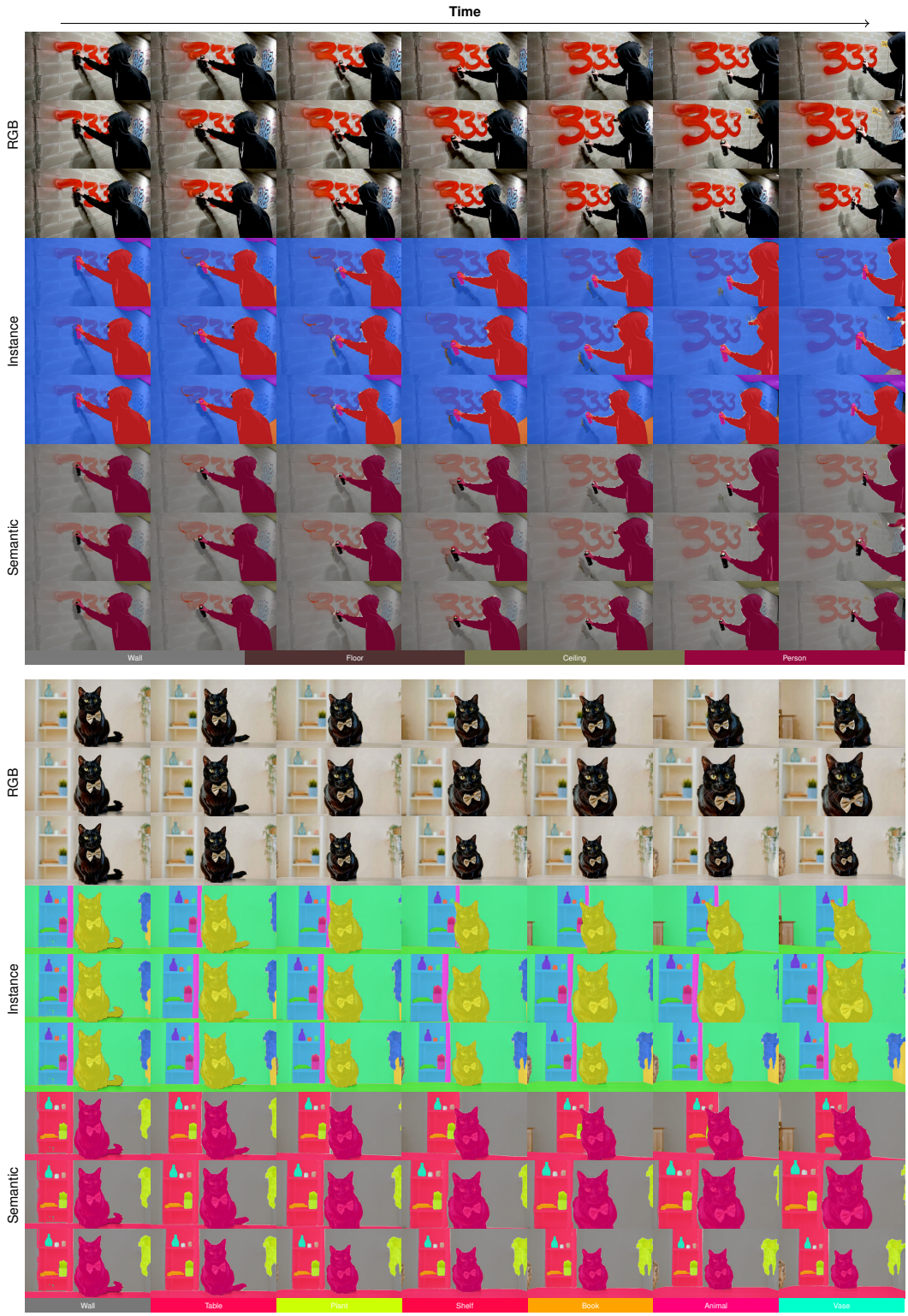


Figure 18: **LucidLyra-SDG dynamic evaluation data.** RGB frames with reference semantic and instance annotations from dynamic evaluation scenes.