RAPP3R: REINFORCING OBJECT RELATIONSHIP FOR PHYSICALLY PLAUSIBLE 3D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

We present RAPP3R, a unified framework that combines architectural enhancements with a reinforcement learning strategy for physically plausible multi-object 3D scene reconstruction from a single image. Building on a pre-trained joint shape-pose generator that processes each object independently, RAPP3R equips the model to reason about interactions across the objects. Gated Inter-Object Attention introduces an inter-object branch while preserving the pre-trained model's behavior at initialization. Volume-RoPE further encodes each object's position and spatial extent, providing explicit spatial context for attention across objects. Building on these architectural changes, we train the model with GDPO to learn physical relationships in a scene, such as collisions and floating. We overcome the limitations of scene-level reinforcement learning through object-wise advantage computation. As a result, our experiments show that RAPP3R outperforms prior methods in both pose accuracy and physical plausibility.

1 INTRODUCTION

Reconstructing real-world scenes requires capturing both object geometry and physically consistent spatial relationships to support applications such as robotic manipulation, physical simulation, and interactive digital twins. Recovering such representations from one or a few images is challenging because occlusion and depth ambiguity leave geometry and placement underdetermined, while the reconstructed scene must remain visually faithful and physically plausible. Recent generative approaches (Chen et al., 2026; Zadaianchuk et al., 2026) recover object shapes and spatial configurations, but object-level plausibility does not ensure scene-level physical consistency. Small placement errors can break contact, leave objects floating, or introduce inter-object penetration, limiting the usefulness of the reconstructed environment.

Physical consistency depends on how objects are arranged relative to one another, motivating explicit modeling of inter-object relationships. Inter-object attention (Huang et al., 2025; Meng et al., 2026) enables information exchange across objects, while physics-aware methods explicitly correct generated scenes through contact-guided rejection sampling (Yu et al., 2026) or joint shape-pose optimization (Huang et al., 2026b). However, these correction procedures require additional scene-specific computation at inference time. To reduce reliance on such correction and incorporate physical consistency into the generator, we turn to reinforcement learning (RL) with geometric and physical rewards.

To this end, we introduce **RAPP3R**, **R**einforcing Object Rel**A**tionship for **P**hysically **P**lausible **3**D **R**econstruction, a unified architecture and training framework for multi-object 3D reconstruction from a single image. RAPP3R couples explicit modeling of inter-object relationships with object-wise reward assignment to learn physically consistent object arrangements. *Gated Inter-Object Attention* enables information exchange across objects while preserving the pretrained model's behavior at initialization. *Volume-RoPE* encodes both object positions and spatial extents, providing spatial context for inter-object attention. We train RAPP3R through reinforcement learning, using geometric and physical rewards inspired by physics-aware optimization (Yu et al., 2026; Huang et al., 2026b). To distinguish the quality of individual objects within a generated scene, we introduce *Object-wise GDPO*, which extends GDPO's reward-wise normalization (Liu et al., 2026) to individual objects and assigns each object's pose update a separate advantage.

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.

Our experiments demonstrate that RAPP3R reconstructs 3D scenes with improved physical consistency and better alignment with the ground truth compared with previous methods. Our main contributions are:

- We adapt a pre-trained joint shape-pose generator to model inter-object relationships through *Gated Inter-Object Attention*, preserving its original behavior at initialization.
- We derive *Volume-RoPE* by averaging rotary positional embeddings over object bounding boxes, encoding both object positions and spatial extents for inter-object attention.
- We develop *Object-wise GDPO* for reinforcement learning with geometric and physical rewards, extending GDPO's reward-wise normalization (Liu et al., 2026) to individual objects to provide object-specific reward assignment.

2 Related Work

2.1 Generative 3D Reconstruction

Generative 3D reconstruction uses learned generative priors to recover 3D geometry and appearance from visual observations. Many image-to-3D generators (Zhang et al., 2023; Li et al., 2025b; Yang et al., 2024; Xiang et al., 2025; 2026; Wu et al., 2025) employ diffusion (Ho et al., 2020) or flow matching (Lipman et al., 2023) in latent spaces and primarily focus on individual objects. Composing these objects into a scene additionally requires predicting their scales, rotations and translations. MegaPose (Labbé et al., 2022), SAM-6D (Lin et al., 2024), and FoundationPose (Wen et al., 2024) estimate these 6D poses using supplied object models or additional reference observations.

To recover geometry and placement within a unified framework, recent generative approaches jointly infer object shapes and spatial configurations from scene observations. SAM3D (Chen et al., 2026), RecGen (Zadaianchuk et al., 2026), and CUPID (Huang et al., 2026a) jointly model canonical object geometry and the relative pose between the object and camera. MIDI (Huang et al., 2025) and SceneGen (Meng et al., 2026) further model inter-object relationships through attention mechanisms to generate spatially coherent scenes. Despite these advances in geometric reconstruction and spatial coherence, physical artifacts such as inter-object penetration and floating objects persist. We address these challenges by aligning multi-object scene generation with geometric and physical objectives through explicit object-wise reward assignment.

2.2 Physics-aware 3D Scene Reconstruction

Physics-aware methods often augment pretrained reconstruction models such as SAM3D (Chen et al., 2026) with additional inference-time sampling or optimization to explicitly enforce physical constraints. PICASSO (Yu et al., 2026) uses an inferred contact graph to guide rejection sampling under contact and non-penetration constraints. SPARCS (Huang et al., 2026b) and CA-World (Dong et al., 2026) incorporate physical constraints or penalties into scene-specific optimization. MV-SAM3D (Li et al., 2026a) applies collision and contact guidance during structure generation in a multi-view setting, followed by global pose refinement.

Other approaches use simulation feedback to guide scene correction. REST3D (Ma et al., 2026) evaluates candidate layouts through simulation within hierarchical optimization, while ϕ-Scene (Li et al., 2026b) optimizes object positions to resolve penetrations and uses rigid-body simulation to settle objects in an order determined by their support relationships. SimuScene (Lee et al., 2026) uses diagnostic simulation to guide gravity-axis shape stretching and amodal shape resampling. Complementing these inference-time approaches, our method uses object-wise geometric and physical rewards to improve the flow-matching generator during training.

2.3 Alignment of 3D Generative Models

Reinforcement learning enables generative models to optimize externally evaluated objectives without differentiating through the evaluator, as demonstrated by large language model alignment with human preferences (Ouyang et al., 2022) and verifiable rewards (Shao et al., 2024). Within this paradigm, Group Relative Policy Optimization (GRPO) (Shao et al., 2024) estimates advantages

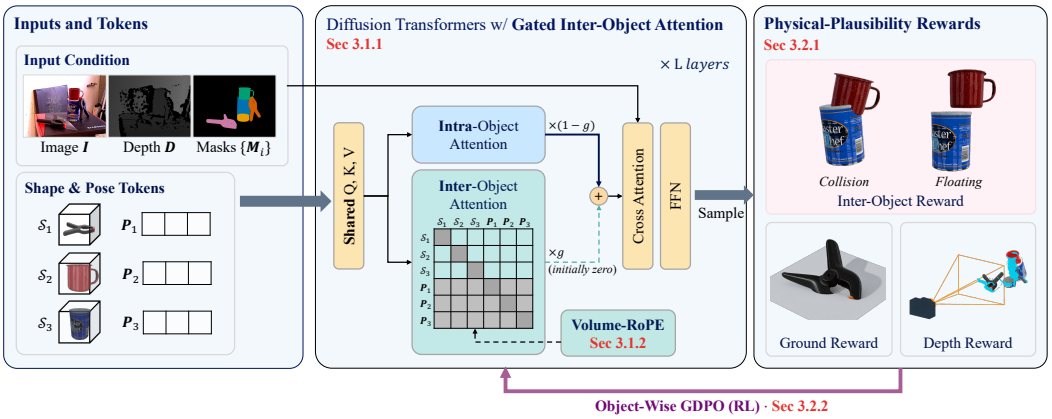


Figure 1: **The overall pipeline of RAPP3R.** RAPP3R integrates gated inter-object attention into a pretrained model to capture object relationships (section 3.1.1), with Volume-RoPE encoding object positions and spatial extents (section 3.1.2). We train the model using geometric and physical rewards (section 3.2.1) with Object-wise GDPO for object-specific reward assignment (section 3.2.2).

from relative rewards among sampled outputs, while Group reward-Decoupled Normalization Policy Optimization (GDPO) (Liu et al., 2026) separately normalizes reward components to accommodate multiple objectives. These approaches have also been extended to denoising models, with Diffusion-DPO (Wallace et al., 2024) adapting Direct Preference Optimization (DPO) (Rafailov et al., 2023) to diffusion models and Flow-GRPO (Liu et al., 2025) enabling online reward optimization for flow matching models through SDE sampling.

For 3D generation, DSO (Li et al., 2025a) adapts Diffusion-DPO (Wallace et al., 2024) to fine-tune single-object generators using simulator-derived stability feedback. SimuScene (Lee et al., 2026) uses flow-matching DPO to improve occlusion-robust amodal shape completion within a simulation-guided reconstruction pipeline. For multi-object generation, PhyMix (Wu et al., 2026) introduces Scene-GRPO, which reweights scene-level flow-matching losses using physics-evaluator rewards and combines this training with test-time refinement. However, Scene-GRPO assigns a shared advantage to all objects within each candidate scene, providing no separate reward for well-placed and poorly placed objects. Our work extends GDPO's reward-wise normalization to individual objects, assigning each object's pose update a separate advantage derived from geometric and physical rewards within a jointly generated scene.

3 Method

RAPP3R jointly generates multiple 3D objects from a single image, recovering their geometries and spatial arrangement. Given an input image $\boldsymbol{I}$, a depth map $\boldsymbol{D}$, and object masks $\{\boldsymbol{M}_i\}_{i=1}^N$, the model predicts each object's shape $\mathcal{S}_i$ and pose $\boldsymbol{P}_i = (\boldsymbol{s}_i, \boldsymbol{R}_i, \boldsymbol{t}_i)$, where $\boldsymbol{s}_i$, $\boldsymbol{R}_i$, and $\boldsymbol{t}_i$ denote scale, rotation, and translation, respectively.

Our method builds on SAM3D (Chen et al., 2026), a pre-trained flow-based joint shape-pose estimator p_{ref} that generates each object's shape and pose independently: *i.e.* $p_{\text{ref}}\big(\{(\mathcal{S},\boldsymbol{P})_i\}_{i=1}^N \mid \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) = \prod_{i=1}^N p_{\text{ref}}(\mathcal{S}_i, \boldsymbol{P}_i \mid \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{M}_i)$. We present a unified framework for model architecture (section 3.1) and training (section 3.2) to improve physical plausibility, with the overall pipeline illustrated in fig. 1.

3.1 Architecture

3.1.1 Gated Inter-Object Attention

To achieve scene-level physical plausibility, it is necessary to consider physical relationships between objects, such as penetration and floating. We therefore introduce inter-object attention inspired by prior work on multi-object generation (Huang et al., 2025; Meng et al., 2026). Following MIDI (Huang et al., 2025), we reuse pre-trained model parameters for inter-object attention. The

object pose tokens in the original SAM3D can attend to the shape and pose tokens of the same object, which we refer to as intra-object attention. In contrast, inter-object attention allows pose tokens to attend to the shape and pose tokens of other objects.

To incorporate inter-object attention into the model, MIDI selectively replaces some intra-object attention layers with their corresponding inter-object attention layers. However, this replacement leads to an initial performance degradation because the layers were originally trained on intra-object attention rather than inter-object attention. Inspired by prior work (Zhang et al., 2024; Blattmann et al., 2023) on attaching new modules to pre-trained models to impart new functionality, we introduce a gating mechanism (named after Zhang et al. 2024) that interpolates between intra-object and inter-object attention. Instead of the original attention, a weighted average of intra-object and inter-object attention is used with shared parameters and a trainable weight $g \in [-1, 1]$. By initializing g to 0, all attention layers initially operate as intra-object attention and are gradually trained to consider inter-object relationships.

3.1.2 VOLUME-ROPE

When shape and pose tokens from multiple objects interact, spatial cues such as adjacency can help the model relate them to one another. Across diverse transformer architectures, positional encodings provide such spatial context. Rotary Positional Encoding (RoPE) (Su et al., 2024) has recently become a widely adopted choice for this purpose.

Conceptually, for a token at position x, RoPE rotates its query and key representations by ωx radians (*i.e.* multiplies them by $e^{i\omega x}$), where ω is the frequency associated with the corresponding vector dimension. A naive extension of RoPE to multi-object 3D generation would use each object's center as its RoPE position; we call this formulation as Center-RoPE (fig. 2). However, center locations alone cannot fully characterize spatial relationships between objects, as relationships such as adjacency also depend on their spatial extents.

We therefore derive Volume-RoPE, which represents both the position and volume of an object. Consider arbitrary points x in object i and y in object j. Under standard RoPE, their relative position is embedded as $e^{i\omega(x-y)}$. We approximate the mean relative embedding between the two objects by averaging over their bounding boxes:

$$\mathbb{E}[e^{i\omega(x-y)}] = \frac{1}{4h_i h_j} \int_{t_j-h_j}^{t_j+h_j} \int_{t_i-h_i}^{t_i+h_i} e^{i\omega(x-y)} \mathrm{d}x\mathrm{d}y = \mathrm{sinc}(\omega h_i)\mathrm{sinc}(\omega h_j) e^{i\omega(t_i-t_j)}, \quad (1)$$

where t_i and h_i denote the center position and half-extent of object i, respectively. Thus, the positional encoding of object i can be represented as $\mathrm{sinc}(\omega h_i)e^{i\omega t_i}$, a computationally feasible form, which we call as Volume-RoPE. The derivation is illustrated intuitively in fig. 2; we defer the mathematical details to section A.1.

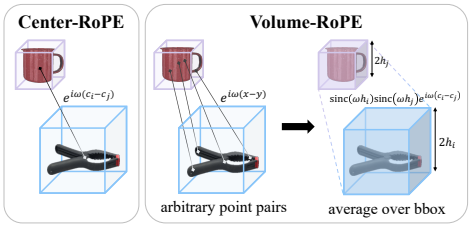


Figure 2: **Volume-RoPE.** Unlike Center-RoPE, Volume-RoPE encodes the spatial extents of generated objects.

A challenge is that Volume-RoPE requires object positions and spatial extents, which are unknown before generation. We approximate these quantities from a pointmap $\mathcal{P}$ derived from the depth map. For each object, we construct a cubic bounding box from the masked pointmap $\boldsymbol{M}_i \odot \mathcal{P}$ and denote its center and half-extent by c_i and h_i, respectively. To achieve invariance to scene-level scaling and translation, we subtract the mean of $\{c_i\}_{i=1}^N$ from the centers and divide both the centered positions and the half-extents by the standard deviation of $\{c_i\}_{i=1}^N$. Using these values, we replace t_i with c_i to obtain the Volume-RoPE encoding $\mathrm{sinc}(\omega h_i)e^{i\omega c_i}$.

3.2 TRAINING

We introduce four types of reward to inject physical plausibility into the pre-trained model: (1) inter-object reward (io), (2) ground reward (grd), (3) depth reward (depth), and (4) GT reward (gt). Throughout this section, the scalar $r_{i,a,b}$ denotes the reward of type $b \in \{\mathrm{io}, \mathrm{grd}, \mathrm{depth}, \mathrm{gt}\}$ for object a in sample i.

3.2.1 PHYSICAL-PLAUSIBILITY REWARDS

Inter-object reward. Let $\mathcal{O}_{i,a}$ denote the posed object a in sample i. The inter-object reward is

$$r_{i,a,\text{io}} = -\frac{1}{N-1}\sum_{\substack{k=1\\k\neq a}}^{N}\delta(\mathcal{O}_{i,a},\mathcal{O}_{i,k}) - \frac{1}{|\mathcal{N}_a|}\sum_{k\in\mathcal{N}_a} d(\mathcal{O}_{i,a},\mathcal{O}_{i,k}). \tag{2}$$

Here, $\delta(A,B)\geq 0$ and $d(A,B)\geq 0$ denote the penetration depth and distance between posed objects A and B, respectively. $\mathcal{N}_a$ contains the indices of objects in contact with object a in the ground-truth contact graph (fig. 3). The two terms penalize inter-object penetration and separation between objects that should be in contact, respectively.

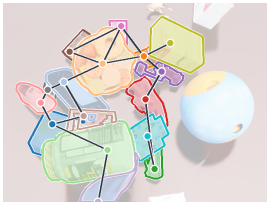


Figure 3: Contact Graph.

Ground reward. Let $z(\mathcal{O}_{i,a}) := \min_{\boldsymbol{p}\in\mathcal{O}_{i,a}} \boldsymbol{p}_z$ denote the lowest vertical coordinate of object a in sample i. Assume that the ground plane is at $z=0$, the ground reward is

$$r_{i,a,\text{grd}} = \min(0, z(\mathcal{O}_{i,a})) - \mathbb{1}_{a\in\mathbb{G}}\max(0, z(\mathcal{O}_{i,a})), \tag{3}$$

where $\mathbb{G}$ indexes objects touching the ground in the ground-truth scene. The first term penalizes ground penetration for every object, while the second penalizes floating for objects in $\mathbb{G}$.

Depth reward. This reward discourages predicted objects from occupying observed empty space. Inspired by the free-space constraint in PICASSO (Yu et al., 2026), the depth reward penalizes object regions that are rendered in front of the observed depth map:

$$r_{i,a,\text{depth}} = -\sum_{\boldsymbol{x}\in\text{render}(\mathcal{O}_{i,a})} \max(0, \boldsymbol{D}_{\boldsymbol{x}} - \hat{\boldsymbol{D}}_{i,\boldsymbol{x}}), \tag{4}$$

where $\boldsymbol{x}$ indexes an image pixel, $\boldsymbol{D}_{\boldsymbol{x}}$ is the observed depth at that pixel, $\hat{\boldsymbol{D}}_{i,\boldsymbol{x}}$ is the depth rendered from the predicted scene in sample i , and $\text{render}(\mathcal{O}_{i,a})$ denotes the set of pixels with valid observed depth at which object a is visible in the rendered scene. When $\hat{\boldsymbol{D}}_{i,\boldsymbol{x}} < \boldsymbol{D}_{\boldsymbol{x}}$, *i.e.* when part of an object lies closer to the camera than the observed surface, the object receives a negative reward.

GT reward. The GT reward regularizes the predicted object poses toward their ground-truth values during RL fine-tuning. It is defined as the negative ℓ_2 distance between the predicted and ground-truth pose parameters, comprising scale $\boldsymbol{s}_a$, 6D rotation representation $\boldsymbol{r}_a$, and translation $\boldsymbol{t}_a$:

$$r_{i,a,\text{gt}} = -\left\|(\log\hat{\boldsymbol{s}}_{i,a}, \hat{\boldsymbol{r}}_{i,a}, \hat{\boldsymbol{t}}_{i,a}) - (\log\boldsymbol{s}_a, \boldsymbol{r}_a, \boldsymbol{t}_a)\right\|_2, \tag{5}$$

3.2.2 OBJECT-WISE GDPO

The equations below describe an application of Flow-GRPO (Liu et al., 2025) to a flow-based pose estimator with conditioning inputs $\mathcal{C} = (\boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_a\}_{a=1}^N, \{\mathcal{S}_a\}_{a=1}^N)$, comprising the input image, depth map, object masks, and ground-truth object shapes:

$$\mathcal{J}(\theta) = \mathbb{E}_{\mathcal{C}\sim\mathcal{D},\{\boldsymbol{x}_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|\mathcal{C})}[f(\theta)], \tag{6}$$

$$f(\theta) = \frac{1}{GT}\sum_{i=1}^{G}\sum_{t=1}^{T}\left[\min\left(s_i^t(\theta)\hat{A}_i,\ \text{clip}_{1-\epsilon}^{1+\epsilon}(s_i^t(\theta))\hat{A}_i\right) - \beta D_{\text{KL}}\left(\pi_\theta\,\|\,\pi_{\text{ref}}\right)\right]. \tag{7}$$

$$s_i^t(\theta) = \frac{p_\theta(\boldsymbol{x}_i^{t-1}\mid\boldsymbol{x}_i^t,\mathcal{C})}{p_{\theta_{\text{old}}}(\boldsymbol{x}_i^{t-1}\mid\boldsymbol{x}_i^t,\mathcal{C})}. \tag{8}$$

Here, $\mathcal{D}$ is the training-data distribution, and π_θ denotes the stochastic denoising policy parameterized by θ, with one-step transition density p_θ. Each $\boldsymbol{x}_i = (\boldsymbol{x}_i^T, \ldots, \boldsymbol{x}_i^0)$ is a scene-level pose-estimation trajectory obtained by rolling out the behavior policy $\pi_{\theta_{\text{old}}}$ for T denoising steps. The objective $\mathcal{J}(\theta)$ is maximized using the group-relative advantage $\hat{A}_i$ computed from G samples. Details of the stochastic policy and KL computation are provided in section A.2.

We fine-tune only the pose-estimation component of the pre-trained model while fixing the shape-generation latent at step t to $\boldsymbol{z}^t = \frac{T-t}{T}\boldsymbol{z}^0 + \frac{t}{T}\boldsymbol{\xi}$, where $\boldsymbol{z}^0$ is the ground-truth shape latent and $\boldsymbol{\xi}\sim\mathcal{N}(\mathbf{0},\mathbf{I})$ is Gaussian noise.

To prevent reward components with larger scales or variances from dominating the learning signal, GDPO (Liu et al., 2026) employs two-step normalization to get the advantage:

$$A_{i,b} = \frac{r_{i,b} - \text{mean}\{r_{j,b}\}_{j=1}^{G}}{\text{std}\{r_{j,b}\}_{j=1}^{G}}, \quad A_i = \sum_{b=1}^{M} A_{i,b}, \quad \hat{A}_i = \frac{A_i - \text{mean}\{A_j\}_{j=1}^{G}}{\text{std}\{A_j\}_{j=1}^{G}}, \tag{9}$$

where b indexes the reward type and M is the number of reward types. GDPO first normalizes rewards within each reward type, sums the normalized rewards across types, and then normalizes the aggregate scores across samples to obtain $\hat{A}_i$.

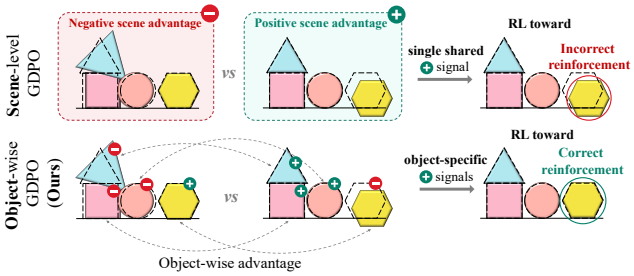


Figure 4: Comparison of Scene-level GDPO and Object-wise GDPO.

However, directly applying GRPO or GDPO to multi-object 3D generation cannot distinguish quality differences among individual objects because all objects share a scene-level advantage, as in the Scene-GRPO formulation of PhyMix (Wu et al., 2026). Figure 4 illustrates how an incorrectly placed object in a scene with a positive advantage can be reinforced, whereas a correctly placed object in a scene with a negative advantage can be penalized.

To address this issue, we extend GDPO's reward-decoupling strategy to the object level:

$$A_{i,a,b} = \frac{r_{i,a,b} - \text{mean}\{r_{j,a,b}\}_{j=1}^{G}}{\text{std}\{r_{j,a,b}\}_{j=1}^{G}}, \quad A_{i,a} = \sum_{b=1}^{M} A_{i,a,b}, \tag{10}$$

where a indexes an object in a scene. The decoupled advantages are then normalized along the group axis:

$$\hat{A}_{i,a} = \frac{A_{i,a} - \text{mean}\{A_{j,a}\}_{j=1}^{G}}{\text{std}\{A_{j,a}\}_{j=1}^{G}}. \tag{11}$$

Finally, we compute the object-wise objective $g(\theta)$ using the object-decoupled advantage $\hat{A}_{i,a}$:

$$g(\theta) = \frac{1}{GT} \sum_{i=1}^{G} \sum_{t=1}^{T} \left\{ \sum_{a=1}^{N} \left[\min\left(s_{i,a}^{t}(\theta)\hat{A}_{i,a}, \text{clip}_{1-\epsilon}^{1+\epsilon}(s_{i,a}^{t}(\theta))\hat{A}_{i,a} \right) \right] - \beta D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}) \right\}. \tag{12}$$

$$s_{i,a}^{t}(\theta) = \frac{p_\theta(\boldsymbol{x}_{i,a}^{t-1} \mid \boldsymbol{x}_i^t, \mathcal{C})}{p_{\theta_{\text{old}}}(\boldsymbol{x}_{i,a}^{t-1} \mid \boldsymbol{x}_i^t, \mathcal{C})}. \tag{13}$$

Replacing $f(\theta)$ with $g(\theta)$ in eq. (6) yields our object-wise RL objective. The final training loss combines this KL-regularized objective with the SFT loss defined in section A.3:

$$\mathcal{L}(\theta) = -\mathcal{J}(\theta) + \lambda \mathcal{L}_{\text{SFT}}(\theta). \tag{14}$$

4 Experiments

4.1 Setup

Datasets. We train on the dataset introduced by FoundationPose (Wen et al., 2024), which comprises synthetic scenes containing multiple objects from GSO (Downs et al., 2022). This dataset is well suited for learning inter-object interactions because it captures diverse physical relationships, including support, contact, and occlusion. We evaluate the generated poses on the YCB-V dataset (Xiang et al., 2017), a widely used benchmark for 6D object pose estimation. The real-world scenes in YCB-V likewise span a broad range of physical configurations and interactions. The benchmark comprises 12 such scenes with 75 frames each, yielding a total of 900 inputs.



Figure 5: **Qualitative Results.** The output 3D scenes are rendered in the original camera view of input images. The parts where our method does not violate physics while others show collision are emphasized.

Training. For GDPO training, we use size of a group $G = 16$. Coefficients of regularization terms in eq. (14) are set to be $\beta = 1.0$ and $\lambda = 1.0$ empirically. During group sampling, we set inference step $T = 10$ following the recipe of Flow-GRPO. While learning rate is basically $\gamma = 4 \times 10^{-4}$, the learning rate for gating coefficient g is 50 times larger so that the model can utilize inter-object attention more quickly. We finetune the model using LoRA (Hu et al., 2022) with rank 32. We trained the model using 500 scene samples of the training data.

Metrics. ADD-S (Xiang et al., 2017) is the most commonly used metric for evaluating pose estimation accuracy. We use the SAM3D variant of ADD-S, which accounts for both shape-generation and pose-estimation accuracy. We also report ADD-S@0.1, defined as the fraction of object predictions with normalized ADD-S below 0.1. To evaluate physical plausibility, we report the average collision ratio (Collision) as a percentage (%) and the average floating distance (Floating) and ground displacement (Ground) in millimeters (mm). The inter-object collision ratio is the fraction of object pairs in which one object penetrates more than 5% of the other. Floating distance is zero for colliding objects and otherwise measured to the nearest object that has a lower center and shares support. For objects touching the ground in the ground-truth scene, ground displacement measures the distance from their bottom to the ground plane.

Baselines. We compare our method against three training-based joint shape-and-pose generation methods: MIDI (Huang et al., 2025), RecGen (Zadaianchuk et al., 2026), and SAM3D (Chen et al., 2026). We additionally evaluate SAM3D combined with the optimization-based method PICASSO (Yu et al., 2026). For a direct comparison under the same refinement protocol, we also apply PICASSO to our method.

4.2 Main Results

Table 1 reports the shape-pose accuracy and physical plausibility of the scenes generated by the baselines and RAPP3R. Compared with SAM3D, our architectural modifications and reinforcement learning reduce the inter-object collision ratio and floating distance by 51% and 55%, respectively, while also improving ADD-S. Moreover, RAPP3R achieves lower ADD-S, collision ratio, and floating distance than all other baselines.

We additionally report the results of applying PICASSO to SAM3D and RAPP3R in table 2. Although PICASSO improves the pose accuracy of both methods, it increases their collision ratios and substantially worsens the floating distance for RAPP3R. These results reveal trade-offs between pose accuracy and physical plausibility under PICASSO refinement. Nevertheless, RAPP3R com-

Table 1: **Results.** Quantitative comparison of our approach and baselines.

Method	Alignment		Physics		
	ADD-S ↓	ADD-S@0.1↑	Collision ↓	Floating ↓	Ground ↓
MIDI (Huang et al., 2025)	0.3372	0.0989	17.23	5.36	4.27
RecGen (Zadaianchuk et al., 2026)	0.0780	0.9329	8.37	2.24	9.97
SAM3D (Chen et al., 2026)	0.0697	0.8962	10.24	3.35	9.47
SAM3D (*repo ver.*)	0.0910	0.7724	18.13	1.63	16.66
RAPP3R (Ours)	0.0685	0.9556	5.01	1.50	11.20

* Explanation about *repo ver.* is in section A.5.

Table 2: **Post-Processing Results.**

Method	Alignment		Physics		
	ADD-S ↓	ADD-S@0.1↑	Collision ↓	Floating ↓	Ground ↓
SAM3D + PICASSO (Yu et al., 2026)	0.0492	0.9707	11.09	**2.90**	7.82
RAPP3R (Ours) + PICASSO	**0.0482**	**0.9782**	**9.34**	4.21	**5.84**

bined with PICASSO outperforms SAM3D with PICASSO on four of the five metrics, suggesting that the initial reconstruction affects refinement outcomes.

Figure 5 complements the quantitative evaluation with qualitative comparisons. In the illustrated scenes, RAPP3R avoids inter-object collisions and maintains physically plausible contact relationships, whereas the baseline reconstructions exhibit collisions or floating artifacts. Together, the qualitative and quantitative results also demonstrate synthetic-to-real transfer, as RAPP3R achieves these improvements on real-world YCB-V (Xiang et al., 2017) inputs after fine-tuning on synthetic FoundationPose (Wen et al., 2024) scenes.

4.3 Ablation Study

4.3.1 Inter-Object Attention

Table 3 evaluates inter-object attention and positional encoding under our proposed RL method. Applying RL alone (SAM3D + RL) reduces collision and floating relative to the pre-trained model, but worsens ADD-S and ground displacement. Adding gated inter-object attention through simple concatenation without positional encoding degrades all reported metrics. This suggests that enabling inter-object information exchange alone is insufficient to improve reconstruction.

Both Center-RoPE and Volume-RoPE improve performance over inter-object attention without positional encoding, supporting the importance of explicit spatial guidance. Although Center-RoPE yields lower collision ratio and ground displacement than Volume-RoPE, it still produces higher ADD-S and more than twice the floating distance of the pre-trained SAM3D model. By contrast, Volume-RoPE achieves the best alignment metrics among all evaluated variants while reducing both collision and floating relative to SAM3D. These results demonstrate the benefit of incorporating object extents into positional encoding for accurate and physically plausible reconstruction.

4.3.2 Object-Wise GDPO

The quantitative results comparing Scene-level GDPO and Object-wise GDPO are reported in table 4. With both approaches learning better GT alignment, Scene-level GDPO scored better in mean ADD-S and Object-wise GDPO in accuracy. Also, both models prevented the floating of the pre-trained model, as demonstrated by the lower floating distance compared to SAM3D. However, the collision ratio of Scene-level GDPO increased from SAM3D, while Object-wise GDPO was successfully trained to reduce collisions by half. These results suggest that Scene-level GDPO is less effective at reducing inter-object collisions, consistent with our hypothesis in section 3.2.2.

Table 3: Ablation Study on **Inter-Object Attention (IOA)** and **Positional Encoding.**

Variants	Alignment		Physics		
	ADD-S ↓	ADD-S@0.1↑	Collision ↓	Floating ↓	Ground ↓
SAM3D	0.0697	0.8962	10.24	3.35	9.47
+ RL	0.0777	0.9469	5.40	1.29	14.73
+ RL + IOA	0.2672	0.0480	13.10	12.43	33.90
+ RL + IOA + Center-RoPE	0.0797	0.8781	4.23	7.95	9.40
+ RL + IOA + Volume-RoPE	0.0685	0.9556	5.01	1.50	11.20

Table 4: Ablation Study on **Object-wise GDPO.**

	Alignment		Physics		
	ADD-S ↓	ADD-S@0.1 ↑	Collision ↓	Floating ↓	Ground ↓
Scene-Level GDPO	**0.0605**	0.9479	10.37	**0.58**	17.03
Object-Wise GDPO	0.0685	**0.9556**	**5.01**	1.50	**11.20**

4.4 Analysis

Gating Factor. The trainable gating parameters g control the contribution of inter-object attention to pose estimation. Figure 6 plots the mean absolute value of g across layers and attention heads over the course of training. As expected, the increasing magnitude of g indicates that the model assigns progressively greater weight to inter-object attention during training. By step 500, the mean magnitude reaches approximately 0.25 and increases further to 0.30 by step 700. Volume-RoPE and Center-RoPE exhibit similar gating dynamics. Without RoPE, however, the magnitude of g grows more slowly than with the RoPE-based variants. This slower growth suggests that learning inter-object interactions is more difficult in the absence of positional encoding.

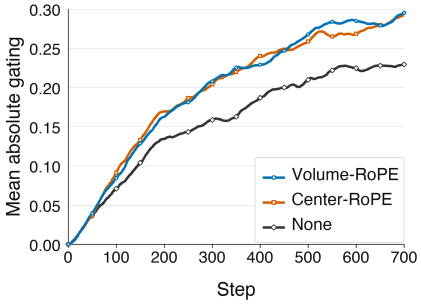


Figure 6: Evolution of the Gating Parameter g

Attention Map. Volume-RoPE prompts inter-object attention to attend more on physically adjacent objects according to its attention map, compared to the inter-object attention without RoPE. The detailed analysis is in section A.4

5 Conclusion

Conclusion. In this paper, we presented RAPP3R, a unified architecture and training framework for physically plausible 3D scene reconstruction with a joint shape-and-pose generator. Our *Gated Inter-Object Attention* models relationships among objects while faithfully preserving the pre-trained model's intra-object attention behavior at initialization, and *Volume-RoPE* provides explicit spatial information. By reinforcing the pre-trained model with *Object-wise GDPO* and *Physical-Plausibility Rewards*, RAPP3R outperforms the evaluated baselines on most metrics for reconstruction accuracy and physical plausibility. We further validated the effectiveness of each proposed architectural and training component through extensive experiments.

Limitation. Although Object-wise GDPO is broadly applicable to various reward functions, we focus only on object collisions and floating as an initial step toward physically plausible 3D reconstruction. Future work could extend the framework to a broader range of physical properties, such as stability, by incorporating additional reward functions. Moreover, although our method generally achieves better physical plausibility than previous approaches, it cannot *guarantee* physically plausible outputs in every case, which remains an important consideration for real-world applications. Finally, the $\mathrm{sinc}(\omega h_i)$ term causes Volume-RoPE to violate an important property of RoPE: norm preservation. Standard RoPE rotates the query and key while preserving their norms, whereas Volume-RoPE does not. Although we observed no issues in our setting, this violation may cause problems when Volume-RoPE is combined with techniques that rely on norm preservation.

AI USE STATEMENT

In this work, we used generative AI tools to polish the manuscript, assist with coding for experiments, and identify some related work. In addition, AI tools assisted with the derivation of mathematical details, while the human authors performed most of the work. The authors reviewed the AI-assisted text and references, verified the mathematical derivations, and tested the AI-assisted experimental code. The authors take responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Sections 3.1 and 3.2 describe the model architecture, physical-plausibility rewards, and Object-wise GDPO training objective. Section 4.1 reports the datasets, key training hyperparameters, and evaluation metrics. Appendices A.1, A.2, and A.3 provide the Volume-RoPE derivation and details of the stochastic flow policy, KL regularization, and supervised fine-tuning objective. Appendix A.5 clarifies the coordinate normalization used in our experiments.

REFERENCES

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Conference on Computer Vision and Pattern Recognition*, 2023.

Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, et al. Sam 3d: 3dfy anything in images. In *Conference on Computer Vision and Pattern Recognition*, 2026.

Xin Dong, Weijian Deng, Lihan Zhang, Tianru Dai, Wenfeng Deng, and Yansong Tang. Caworld: Multi-object counterfactual alignment for efficient interactive-ready reconstruction. *arXiv preprint arXiv:2605.30239*, 2026.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *International Conference on Robotics and Automation*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Binbin Huang, Haobin Duan, Yiqun Zhao, Zibo Zhao, Yi Ma, and Shenghua Gao. Cupid: Generative 3d reconstruction via joint object and pose modeling. *Conference on Computer Vision and Pattern Recognition*, 2026a.

Wei-Cheng Huang, Jiaheng Han, Xiaohan Ye, Zherong Pan, and Kris Hauser. Simulation-ready cluttered scene estimation via physics-aware joint shape and pose optimization. *Robotics: Science and Systems*, 2026b.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. 2026c.

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

Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. Megapose: 6d pose estimation of novel objects via render & compare. In *Conference on Robot Learning*, 2022.

Inhee Lee, Sangwon Baik, Sungjoo Kim, Hyeonwoo Kim, Hyunsoo Cha, and Hanbyul Joo. Simuscene: Simulation-ready compositional 3d scene reconstruction from a single image. *arXiv preprint arXiv:2606.03994*, 2026.

Baicheng Li, Dong Wu, Jun Li, Shunkai Zhou, Zecui Zeng, Lusong Li, and Hongbin Zha. Mv-sam3d: Adaptive multi-view fusion for layout-aware 3d generation. *arXiv preprint arXiv:2603.11633*, 2026a.

Haodong Li, Lulu Shao, Haolin Lu, Yu Fu, Yen-Ru Chen, Seemandhar Jain, and Manmohan Chandraker. ϕ-scene: Physically grounded image-to-3d scene reconstruction. *arXiv preprint arXiv:2606.21596*, 2026b.

Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. Dso: Aligning 3d generators with simulation feedback for physical soundness. In *International Conference on Computer Vision*, 2025a.

Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *arXiv preprint arXiv:2502.06608*, 2025b.

Jiehong Lin, Lihua Liu, Dekun Lu, and Kui Jia. Sam-6d: Segment anything model meets zero-shot 6d object pose estimation. In *Conference on Computer Vision and Pattern Recognition*, 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. In *Advances in Neural Information Processing Systems*, 2025.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, et al. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. In *International Conference on Machine Learning*, 2026.

Xiaoxuan Ma, Jiashun Wang, Nicolás Ugrinovic, Yehonathan Litman, and Kris Kitani. REST3D: Reconstructing physically stable 3d scenes from a single image. 2026.

Yanxu Meng, Haoning Wu, Ya Zhang, and Weidi Xie. Scenegen: Single-image 3d scene generation in one feedforward pass. In *International Conference on 3D Vision*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 2022.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, 2023.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Conference on Computer Vision and Pattern Recognition*, 2024.

Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Conference on Computer Vision and Pattern Recognition*, 2024.

Dongli Wu, Jingyu Hu, Ka-Hei Hui, Xiaobao Wei, Chengwen Luo, Jianqiang Li, and Zhengzhe Liu. Phymix: Towards physically consistent single-image 3d indoor scene generation with implicit–explicit optimization. *arXiv preprint arXiv:2604.10125*, 2026.

Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Yikang Yang, Yajie Bao, Jiachen Qian, Siyu Zhu, Philip Torr, Xun Cao, and Yao Yao. Direct3d-s2: Gigascale 3d generation made easy with spatial sparse attention. *arXiv preprint arXiv:2505.17412*, 2025.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *Conference on Computer Vision and Pattern Recognition*, 2025.

Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, et al. Native and compact structured latents for 3d generation. In *Conference on Computer Vision and Pattern Recognition*, 2026.

Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *Robotics: Science and Systems*, 2017.

Xianghui Yang, Huiwen Shi, Bowen Zhang, Fan Yang, Jiacheng Wang, Hongxu Zhao, Xinhai Liu, Xinzhou Wang, Qingxiang Lin, Jiaao Yu, Lifu Wang, Jing Xu, Zebin He, Zhuo Chen, Sicong Liu, Junta Wu, Yihang Lian, Shaoxiong Yang, Yuhong Liu, Yong Yang, Di Wang, Jie Jiang, and Chunchao Guo. Hunyuan3D 1.0: A unified framework for text-to-3D and image-to-3D generation. *arXiv preprint arXiv:2411.02293*, 2024.

Xihang Yu, Rajat Talak, Lorenzo Shaikewitz, and Luca Carlone. Picasso: Holistic scene reconstruction with physics-constrained sampling. *Robotics: Science and Systems*, 2026.

Andrii Zadaianchuk, Leonardo Barcellona, Lennard Schuenemann, Christian Gumbsch, Zehao Wang, Muhammad Zubair Irshad, Fabien Despinoy, Rahaf Aljundi, Stratis Gavves, and Sergey Zakharov. Reconstruction by generation: 3d multi-object scene reconstruction from sparse observations. In *European Conference on Computer Vision*, 2026.

Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions on Graphics*, 42(4):1–16, 2023.

Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. In *International Conference on Learning Representations*, 2024.

A APPENDIX

A.1 VOLUME-ROPE: DERIVATION AND IMPLEMENTATION

In this section, mathematical details of section 3.1.2 is explained.

The equation below is a derivation of eq. (1). Using the substitutions $u = x - t_i$ and $v = y - t_j$, we obtain

$$\begin{aligned}
&\frac{1}{4h_ih_j}\int_{t_j-h_j}^{t_j+h_j}\int_{t_i-h_i}^{t_i+h_i} e^{i\omega(x-y)}\mathrm{d}x\mathrm{d}y \\
&= \frac{1}{4h_ih_j}\left(\int_{t_i-h_i}^{t_i+h_i} e^{i\omega x}\mathrm{d}x\right)\left(\int_{t_j-h_j}^{t_j+h_j} e^{-i\omega y}\mathrm{d}y\right) \\
&= \frac{e^{i\omega(t_i-t_j)}}{4h_ih_j}\left(\int_{-h_i}^{h_i} e^{i\omega u}\mathrm{d}u\right)\left(\int_{-h_j}^{h_j} e^{-i\omega v}\mathrm{d}v\right) \\
&= \frac{e^{i\omega(t_i-t_j)}}{4h_ih_j}\left(\frac{e^{i\omega h_i}-e^{-i\omega h_i}}{i\omega}\right)\left(\frac{e^{-i\omega h_j}-e^{i\omega h_j}}{-i\omega}\right) \\
&= \frac{e^{i\omega(t_i-t_j)}}{4h_ih_j}\left(\frac{2\sin(\omega h_i)}{\omega}\right)\left(\frac{2\sin(\omega h_j)}{\omega}\right) \\
&= \frac{\sin(\omega h_i)}{\omega h_i}\frac{\sin(\omega h_j)}{\omega h_j}e^{i\omega(t_i-t_j)} \\
&= \mathrm{sinc}(\omega h_i)\mathrm{sinc}(\omega h_j)e^{i\omega(t_i-t_j)},
\end{aligned} \tag{15}$$

where $\mathrm{sinc}(z) = \sin(z)/z$ for $z \neq 0$ and $\mathrm{sinc}(0) = 1$ by continuity. Consequently, the mean relative embedding factorizes as

$$\left(\mathrm{sinc}(\omega h_i)e^{i\omega t_i}\right)\overline{\left(\mathrm{sinc}(\omega h_j)e^{i\omega t_j}\right)}, \tag{16}$$

which yields the Volume-RoPE embedding $\mathrm{sinc}(\omega h_i)e^{i\omega t_i}$ for object i.

For the practical formulation described in section 3.1.2, we replace the translation t_i with the bounding-box center c_i estimated from the masked pointmap. We then standardize $\{c_i\}_{i=1}^N$ and $\{h_i\}_{i=1}^N$ using the mean and standard deviation of $\{c_i\}_{i=1}^N$; hereafter, c_i and h_i denote these standardized quantities.

To apply Volume-RoPE to real-valued queries and keys, we represent each complex multiplication using a two-dimensional block. Let $d = 6M$ be the query and key dimension, where M is the number of frequencies assigned to each spatial axis. We first define the two-dimensional rotation matrix

$$\boldsymbol{R}_{\mathrm{2D}}(\theta) = \begin{bmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{bmatrix}. \tag{17}$$

For frequency ω_m and axis $a \in \{x, y, z\}$, the Volume-RoPE block for object i is

$$\boldsymbol{V}_{i,m,a} = \mathrm{sinc}(\omega_m h_i^a)\boldsymbol{R}_{\mathrm{2D}}(\omega_m c_i^a). \tag{18}$$

Stacking these blocks in the order $(1,x),(1,y),(1,z),\ldots,(M,x),(M,y),(M,z)$ gives

$$\boldsymbol{V}_i = \bigoplus_{m=1}^{M}\bigoplus_{a\in\{x,y,z\}} \boldsymbol{V}_{i,m,a}, \tag{19}$$

where $\bigoplus$ denotes block-diagonal concatenation. Volume-RoPE is then applied to the query and key vectors as

$$\widetilde{\boldsymbol{q}}_i = \boldsymbol{V}_i\boldsymbol{q}_i, \qquad \widetilde{\boldsymbol{k}}_i = \boldsymbol{V}_i\boldsymbol{k}_i. \tag{20}$$

Because $\boldsymbol{V}_i$ is block diagonal, eq. (20) acts independently on each frequency–axis block: $\widetilde{\boldsymbol{q}}_{i,m,a} = \boldsymbol{V}_{i,m,a}\boldsymbol{q}_{i,m,a}$ and $\widetilde{\boldsymbol{k}}_{j,m,a} = \boldsymbol{V}_{j,m,a}\boldsymbol{k}_{j,m,a}$. Substituting these definitions into the dot product and using $(\boldsymbol{A}\boldsymbol{x})^\top = \boldsymbol{x}^\top\boldsymbol{A}^\top$ gives

$$\begin{aligned}
\widetilde{\boldsymbol{q}}_{i,m,a}^\top\widetilde{\boldsymbol{k}}_{j,m,a} &= \left(\boldsymbol{V}_{i,m,a}\boldsymbol{q}_{i,m,a}\right)^\top\left(\boldsymbol{V}_{j,m,a}\boldsymbol{k}_{j,m,a}\right) \\
&= \boldsymbol{q}_{i,m,a}^\top\boldsymbol{V}_{i,m,a}^\top\boldsymbol{V}_{j,m,a}\boldsymbol{k}_{j,m,a},
\end{aligned} \tag{21}$$

The matrix $\boldsymbol{V}_{i,m,a}^{\top}\boldsymbol{V}_{j,m,a}$ in eq. (21) has a direct interpretation. Substituting eq. (18) and using $\boldsymbol{R}_{2\mathrm{D}}(\alpha)^{\top}\boldsymbol{R}_{2\mathrm{D}}(\beta) = \boldsymbol{R}_{2\mathrm{D}}(\beta-\alpha)$ gives

$$\boldsymbol{V}_{i,m,a}^{\top}\boldsymbol{V}_{j,m,a} = \mathrm{sinc}(\omega_m h_i^a)\mathrm{sinc}(\omega_m h_j^a)\boldsymbol{R}_{2\mathrm{D}}\left(\omega_m(c_j^a - c_i^a)\right). \tag{22}$$

Thus, eq. (21) is a standard relative-position RoPE interaction weighted by the product of the two objects' extent-dependent sinc terms. Under the column-vector convention, the relative rotation $\omega_m(c_j^a - c_i^a)$ corresponds to the complex phase $e^{i\omega_m(c_i^a - c_j^a)}$ in eq. (1).

A.2 Stochastic Flow Policy and KL Regularization

We review how the deterministic flow sampler defines stochastic π_θ and why $D_{\mathrm{KL}}(\pi_\theta\|\pi_{\mathrm{ref}})$ admits a closed form, following Flow-GRPO (Liu et al., 2025). The original flow sampler is deterministic, so Flow-GRPO converts it into an equivalent reverse-time SDE to obtain a stochastic policy for RL. Applying Euler–Maruyama discretization to this SDE gives

$$\boldsymbol{x}_{t+\Delta t} = \boldsymbol{x}_t + \left[\boldsymbol{v}_\theta(\boldsymbol{x}_t, t, \mathcal{C}) + \frac{\sigma_t^2}{2t}\left(\boldsymbol{x}_t + (1-t)\boldsymbol{v}_\theta(\boldsymbol{x}_t, t, \mathcal{C})\right)\right]\Delta t + \sigma_t\sqrt{\Delta t}\,\boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I}), \tag{23}$$

where $\boldsymbol{v}_\theta$ is the velocity predicted by the trainable flow model, Δt is the discretization step, and $\mathcal{C}$ denotes the conditioning inputs. Following Flow-GRPO, we set the noise schedule to

$$\sigma_t = \alpha\sqrt{\frac{t}{1-t}}, \tag{24}$$

where the scalar hyperparameter α controls the amount of stochasticity during sampling. We set $\alpha = 0.3$ in our experiments. Conditioned on $\boldsymbol{x}_t$ and $\mathcal{C}$, the only random term in eq. (23) is $\sigma_t\sqrt{\Delta t}\,\boldsymbol{\epsilon}$. Therefore, denoting all remaining terms by $\bar{\boldsymbol{x}}_{t+\Delta t,\theta}$, the one-step policy is the isotropic Gaussian

$$\pi_\theta(\boldsymbol{x}_{t+\Delta t} \mid \boldsymbol{x}_t, \mathcal{C}) = \mathcal{N}\big(\bar{\boldsymbol{x}}_{t+\Delta t,\theta}, \sigma_t^2\Delta t\,\mathbf{I}\big). \tag{25}$$

The reference policy replaces $\boldsymbol{v}_\theta$ with $\boldsymbol{v}_{\mathrm{ref}}$ while retaining the same covariance. For two Gaussian distributions with the same covariance, the KL divergence depends only on the difference between their means. In this case, it has the closed form

$$D_{\mathrm{KL}}(\pi_\theta\|\pi_{\mathrm{ref}}) = \frac{\|\bar{\boldsymbol{x}}_{t+\Delta t,\theta} - \bar{\boldsymbol{x}}_{t+\Delta t,\mathrm{ref}}\|_2^2}{2\sigma_t^2\Delta t} \tag{26}$$

$$= \frac{\Delta t}{2}\left(\frac{\sigma_t(1-t)}{2t} + \frac{1}{\sigma_t}\right)^2\|\boldsymbol{v}_\theta(\boldsymbol{x}_t, t, \mathcal{C}) - \boldsymbol{v}_{\mathrm{ref}}(\boldsymbol{x}_t, t, \mathcal{C})\|_2^2. \tag{27}$$

A.3 Trajectory-Based Supervised Fine-Tuning

This section details the formulation of $\mathcal{L}_{\mathrm{SFT}}(\theta)$ in eq. (14). Given the ground-truth poses $\boldsymbol{x}_{\mathrm{gt}}$ for a scene, the flow-matching objective (Lipman et al., 2023) trains the model by minimizing the following velocity-prediction error:

$$\mathbb{E}_{t\sim\mathcal{U}(0,1),\,\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\|\boldsymbol{v}_\theta\left((1-t)\boldsymbol{x}_{\mathrm{gt}} + t\boldsymbol{\epsilon}, t, \mathcal{C}\right) - (\boldsymbol{\epsilon} - \boldsymbol{x}_{\mathrm{gt}})\|_2^2\right]. \tag{28}$$

Rather than training v_θ on $((1-t)\boldsymbol{x}_{\mathrm{gt}} + t\boldsymbol{\epsilon}, t, \mathcal{C})$, we train it on $(\boldsymbol{x}_i^t, \frac{t}{T}, \mathcal{C})$, where each $\boldsymbol{x}_i$ is sampled as in eq. (6). This approach offers two benefits. First, the $\boldsymbol{v}_\theta$ computations performed during reinforcement learning can be reused, improving computational efficiency. Second, training on the sampled trajectories makes the model more robust to error accumulation and trajectory shifts induced by reinforcement learning. This rationale parallels those of DAgger (Ross et al., 2011) in imitation learning or Self Forcing (Huang et al., 2026c) in video diffusion. For a sampled state $\boldsymbol{x}_i^t$, whose normalized time is $\frac{t}{T}$, the target velocity is $\frac{T}{t}(\boldsymbol{x}_i^t - \boldsymbol{x}_{\mathrm{gt}})$, which reaches $\boldsymbol{x}_{\mathrm{gt}}$ when integrated backward to time 0. Consequently, averaging over the G sampled trajectories and their T predefined time steps, the SFT loss in eq. (14) is given by

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = \frac{1}{GT}\sum_{i=1}^{G}\sum_{t=1}^{T}\left\|\boldsymbol{v}_\theta\left(\boldsymbol{x}_i^t, \frac{t}{T}, \mathcal{C}\right) - \frac{T}{t}\left(\boldsymbol{x}_i^t - \boldsymbol{x}_{\mathrm{gt}}\right)\right\|_2^2. \tag{29}$$

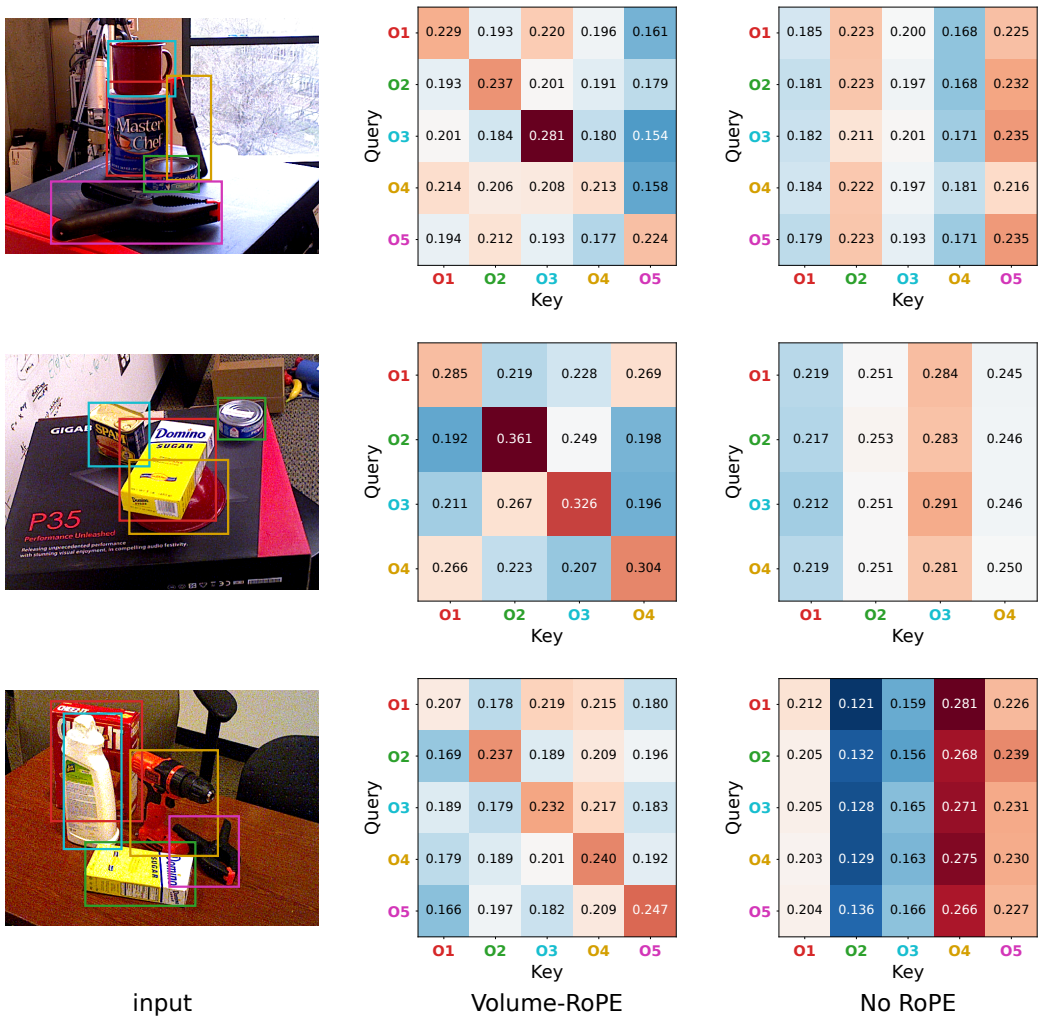


Figure 7: **Average Inter-Object Attention Maps.** Each attention value is averaged across heads and layers, and the attention weights are summed over all tokens associated with each object. These attention maps are sampled at the first time step. Each object is matched to its corresponding query and key using its bounding box color. Red regions indicate higher attention weights, whereas blue regions indicate lower weights.

A.4 Attention Map Analysis

Figure 7 illustrates how the objects attend to one another. In the Volume-RoPE attention maps, each object generally focuses on nearby objects, although this pattern does not hold in every case. For example, in the first input, object 5 focuses on object 2, its nearest neighbor. Object 4 attends to objects 1–4 while assigning less attention weight to object 5, which is relatively far away. In the second input, objects 1 and 4 attend to each other as object 1 is supported by object 4 in the input. Finally, in the third input, object 1 assigns high attention weights to objects 3 and 4, which are also close to object 1. In contrast, in the attention maps of the No RoPE variant, all queries tend to attend to same keys regardless of their spatial arrangement. Together with the results in table 3, this observation suggests that the model has difficulty attending to the appropriate object without an explicit spatial cue such as Volume-RoPE.

A.5 Coordinate Normalization in SAM3D

Because our experiments use a SAM3D normalization setting that differs from the default in the official repository, we clarify this distinction here. In the official SAM3D implementation, the conditioning pointmap is normalized using the scale of the entire scene, such that the predicted scale and translation are expressed in this scene-level coordinate system. In our setup, however, normalizing the input point map independently for each object yields substantially better performance, as shown in table 1, where *repo ver.* denotes the original scene-normalized implementation, while the standard SAM3D row uses object-wise normalization. We also find that SAM3D trains more robustly with reinforcement learning when using the coordinate system induced by object-wise normalization. This normalization scheme is not our contribution; it is already supported by the original SAM3D code through the setting $use_scene_scale = false$.