PAIR3D: LEARNING PHYSICS-ALIGNED AND VISUALLY CONSISTENT 3D SCENE LAYOUTS FROM VERIFIABLE REWARDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-image 3D reconstruction can produce visually plausible scenes, yet reconstructed objects often interpenetrate, float, or become unstable under simulation. Existing physics-based refinement methods solve each scene independently through costly search or hand-designed corrections. We introduce PAIR3D, a reinforcement-learning framework that learns physics-aligned and visually consistent refinement of 3D scene layouts from verifiable rewards. PAIR3D represents a scene using visual evidence, object states, and pairwise relations, and sequentially updates object layouts based on feedback from rendering and physics simulation. This formulation requires neither unique ground-truth layouts nor per-scene optimization and transfers the learned refinement behavior to unseen scenes. Under a matched SAM3D initialization, PAIR3D achieves a 97% scene-level physical success rate while improving visual and geometric consistency. It requires only 4.57 seconds and 24.9 simulator evaluations per scene, substantially reducing the test-time cost of simulation-based search.

1 INTRODUCTION

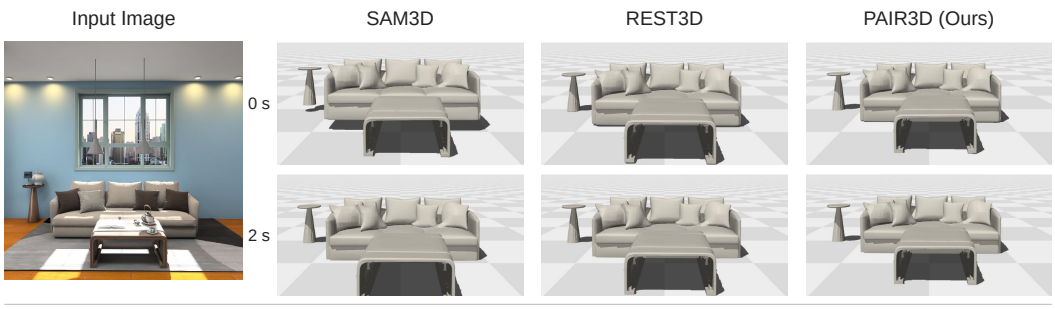
Recent advances in single-image 3D reconstruction can recover a scene as a set of visually faithful objects (Ardelean et al., 2025; Huang et al., 2025; Zhao et al., 2025; Yao et al., 2025). However, visual fidelity does not ensure physical validity. Reconstructed objects may interpenetrate, float above supporting surfaces, or become unstable under gravity. Such failures limit the use of reconstructed scenes in physics simulation, embodied learning, and interactive content creation (Ma et al., 2026; Lee et al., 2026).

Physics simulation provides a natural way to detect such errors. REST3D (Ma et al., 2026) searches over object poses using simulation feedbacks, while SimuScene (Lee et al., 2026) converts simulated dynamics into pose and geometry corrections. Both methods initialize the scene using the image-to-3D model SAM3D. Although effective, their refinement is performed independently for every test scene. REST3D requires costly per-scene search over candidate layouts, while SimuScene follows a sequential correction procedure in which earlier decisions cannot be revisited. Consequently, the experience acquired while repairing one scene is not transferred to the next, potentially resulting in substantial test-time computation and excessive simulator evaluations.

We instead study *physics alignment*: learning to adapt reconstructed scene layouts so that they satisfy simulator-verifiable physical constraints while remaining consistent with the visual observation. This objective differs from simply seeking a stable configuration. Physics alone is under-constrained, as many stable layouts may not resemble the input image. Conversely, direct supervision with a prescribed target layout is difficult, because physically valid solutions are generally non-unique and paired correction targets are costly to obtain. The key observation is that, even when the correct layout is unknown, the consequence of a layout update can be evaluated through physical simulation.

Based on this observation, we introduce PAIR3D, short for *Physics-Aligned and Image-consistent Refinement for 3D scenes*. PAIR3D formulates scene refinement as a sequential decision-making problem and amortizes the refinement process into a reusable reinforcement-learning (RL) policy. Starting from an object-wise reconstruction, the policy iteratively updates the scene. Each scene is represented as a *multimodal relational graph* that combines target-view evidence, structured

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.



	SAM3D	REST3D	PAIR3D (Ours)
Scene mIoU at 0 s (%) ↑	34.14	37.69	**38.20**
Max. displacement, 0–2 s (mm) ↓	136.75	17.91	**13.26**
Max. support error, 0 s (mm) ↓	43.65	7.35	**$<10^{-5}$**
Simulation Calls, 0 s (count) ↓	—	63490	**20**

Figure 1: Physics alignment and visual consistency under various methods for a reconstructed scene. We compare SAM3D (Chen et al., 2026), REST3D (Ma et al., 2026), and PAIR3D (ours) at initialization and after 2 seconds of MuJoCo simulation, rendered from the input camera. In this example, PAIR3D achieves the highest scene mIoU, near-zero support error, and the smallest simulation-induced displacement, while using fewer simulation rollouts for scene refinement. The mIoU and support error are measured at initialization; displacement denotes the maximum per-object translation between 0 and 2 seconds.

object states, and pairwise physical relations. This representation allows the policy to identify both visually inconsistent objects and the interactions responsible for physical violations. After each action, rendering measures mask and depth agreement with the input observation, while geometric checks and rigid-body simulation measure penetration, support, tilt, and residual motion. Reductions in these verifier costs provide dense rewards, and a success reward is issued only when the layout satisfies both physical and visual criteria.

Experiments under a matched SAM3D initialization demonstrate that PAIR3D improves physical validity without sacrificing reconstruction fidelity. It achieves a 97% scene-level success rate across five physical criteria and an mIoU of 47.25%. PAIR3D also reduces the average refinement time from 43.16 s for REST3D to 4.57 s and the number of simulation rollouts from approximately 63.3K to 24.9. Figure 1 illustrates how PAIR3D preserves the observed layout while reducing support errors and simulation-induced motion.

Our contributions are threefold:

- We introduce physics alignment as the problem of adapting reconstructed 3D scene layouts to simulator-verifiable physical constraints while preserving consistency with the input observation.
- We formulate physics alignment as sequential decision making and develop PAIR3D, a multimodal relational graph based policy trained from physical and visual evidences.
- We jointly evaluate physical validity, visual consistency, and test-time efficiency, showing substantial improvements in physical success and simulation cost over per-scene refinement methods under a matched reconstruction setting.

2 Related Work

Compositional 3D reconstruction from a single image. Recent single-image methods increasingly recover a scene as a collection of separately addressable objects rather than a single fused surface. Gen3DSR decomposes the problem into holistic scene parsing and object-level reconstruction, then places reconstructed objects back into a common scene frame (Ardelean et al., 2025). MIDI extends image-to-3D generation to multiple instances with cross-instance attention (Huang et al., 2025). DepR uses depth throughout both object generation and layout optimization to improve scene alignment (Zhao et al., 2025). CAST reconstructs components independently and aligns them with inferred inter-object relations and geometric constraints (Yao et al., 2025). More recent systems such as

3D-RE-GEN and SceneGen further explore compositional generation, object placement, and feed-forward multi-object reconstruction (Sautter et al., 2025; Meng et al., 2026). SAM 3D provides a strong image-to-3D baseline that jointly models object geometry and layout in cluttered images (Chen et al., 2026). These methods make scene objects explicit and manipulable, which provides a useful interface for physical reasoning.

Physics-aware reconstruction and scene estimation. Physical constraints have been incorporated into reconstruction through differentiable simulation, contact reasoning, and joint shape–pose optimization. Representative methods include PhyRecon for multi-view neural reconstruction (Ni et al., 2024), PhysGen3D for single-image interactive scene recovery (Chen et al., 2025), Picasso for physics-constrained multi-object sampling (Yu et al., 2026), and physics-aware joint shape and pose optimization for RGB-D scenes (Huang et al., 2026). Together, these works show that physical reasoning can resolve ambiguities beyond appearance cues, although they differ in observations and optimized scene variables. REST3D (Ma et al., 2026) and SimuScene (Lee et al., 2026) are closest to our setting. REST3D performs support-aware, scene-specific search with the cross-entropy method, while SimuScene diagnoses physical failures and applies predefined pose, scale, or shape corrections. In contrast, PAIR3D learns a reusable policy that maps the current scene state and target observation directly to layout actions.

Learning iterative alignment from feedback. A related line of work learns how to perform iterative optimization. Learning to Optimize casts update selection itself as a policy-learning problem (Li & Malik, 2017). ReAgent applies imitation and reinforcement learning to iterative point-cloud registration, where actions progressively align a model to an observation (Bauer et al., 2021). SporeAgent extends this idea to multi-object pose refinement and uses scene-level physical plausibility to reduce ambiguous placements (Bauer et al., 2022). In generation, DSO uses rigid-body simulation scores to align image-to-3D generators toward physically sound object shapes with preference- and reward-based optimization (Li et al., 2025). Physics-aware scene generation methods such as PhyScene and PAT3D also integrate collision, support, or rigid-body simulation into the synthesis process (Yang et al., 2024; Lin et al., 2026). These works establish that physical feedback can supervise learned 3D systems. Our setting differs in the quantity being learned: we do not learn a new object generator or recover the pose of a known CAD model. Instead we learn a scene-level layout policy for already generated multi-object reconstructions.

3 METHOD

Given an initial object-wise 3D reconstruction, our goal is to adjust the scene layout with physical constraints while preserving the visual evidence that produced it. Following recent work, we use SAM 3D to provide the initial reconstruction.

Three observations have shaped our design. First, it is sequential: each layout edit changes the state on which later edits operate. Second, it is relational: support, penetration, and contact depend on interactions between objects rather than on isolated object states. Third, physical validity alone is insufficient, because a stable scene may no longer agree with the input observation. We therefore formulate the adjustment as closed-loop decision making with three stages: *represent* the current scene using multimodal object and relational evidence, *act* on this representation with a structured policy, and *verify* the resulting layout through executable rendering and physical simulation. Figure 2 summarizes the pipeline.

3.1 PHYSICAL ALIGNMENT AS SEQUENTIAL DECISION MAKING

Let $\{\mathcal{M}_i\}_{i=1}^N$ be the reconstructed meshes. At step t, the scene layout is

$$\ell_i^t = (\mathbf{R}_i^t, \mathbf{t}_i^t, \mathbf{s}_i^t), \qquad \mathcal{L}_t = \{\ell_i^t\}_{i=1}^N, \tag{1}$$

where $\mathbf{R}_i^t$, $\mathbf{t}_i^t$, and $\mathbf{s}_i^t$ denote rotation, translation, and scale in a common scene frame. The target evidence $\mathcal{O}^\star$ consists of instance masks, estimated depth, and camera information. Keeping the meshes fixed separates layout alignment from geometry generation and prevents changes to object shape from trivially satisfying the physical verifier.

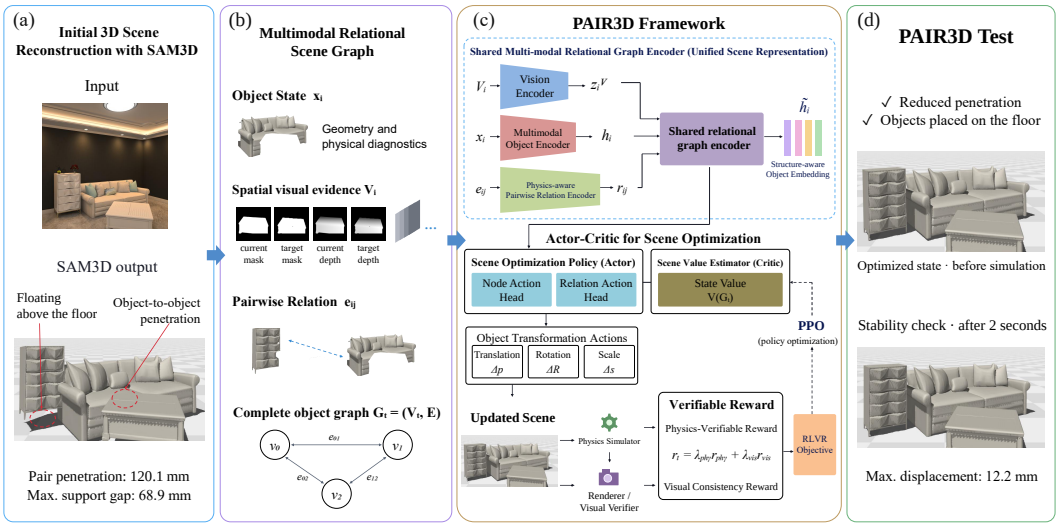


Figure 2: Overview of PAIR3D. PAIR3D aligns reconstructed 3D scenes with physical constraints through verifiable feedback. Given an initial SAM3D reconstruction with artifacts such as floating objects and inter-object penetration, we construct a multimodal relational scene graph from object states, spatial visual evidence, and pairwise relations. A shared relational encoder provides structure-aware representations to an actor–critic policy, which iteratively updates object translation, rotation, and scale. Physics simulation and visual verification provide verifiable rewards for PPO training. The resulting policy directly refines scenes at test time, reducing penetration and support violations while producing configurations that remain stable under simulation.

A reconstruction may admit multiple physically acceptable layouts, so we do not assume a unique target pose for every object. Instead, the policy learns from the verified consequences of sequential actions:

$$a_t \sim \pi_\theta(\cdot \mid \mathcal{G}_t), \qquad (s_{t+1}, \mathbf{b}_t) = \mathrm{Step}_\Xi(s_t, a_t), \tag{2}$$

where s_t contains the current layout and episode state, $\mathcal{G}_t$ is the policy observation, and $\mathbf{b}_t$ contains measurements produced by a fixed verification protocol Ξ. A non-stop action applies a bounded layout transformation to a selected object, optionally conditioned on an object pair. The resulting candidate is rendered and simulated in MuJoCo (Todorov et al., 2012); simulation provides physical measurements but does not replace the policy state with the settled simulator pose. This closed loop exposes contact and dynamic consequences without differentiating through the simulator.

3.2 Multimodal Relational Scene Representation

At each decision step, we organize visual, geometric, and physical evidence into a multimodal relational scene graph,

$$\mathcal{G}_t = \left(\mathcal{V}, \mathcal{E}, \{x_i^t, V_i^t\}_{i\in\mathcal{V}}, \{e_{ij}^t\}_{(i,j)\in\mathcal{E}}\right), \qquad \mathcal{E} = \{(i,j) : i < j\}. \tag{3}$$

Each node combines a structured object descriptor x_i^t with spatial visual evidence V_i^t. The structured descriptor summarizes the current layout together with geometric and physical diagnostics such as image-space discrepancy, support/contact state, and simulated motion. The visual tensor stacks aligned target/current masks and normalized target/current depth, preserving where silhouette and depth discrepancies occur rather than reducing them to a scalar score. Each edge descriptor e_{ij}^t records pairwise evidence such as relative pose, object extents, penetration, and image overlap. The graph therefore anchors individual objects to the observation while exposing the relations that determine physical compatibility.

We use a complete graph rather than a contact-only topology so that potentially relevant pairs remain available before contact occurs. Neither the connectivity nor the descriptors require semantic category labels or category-pair rules; shared functions operate on measured geometry and physical state across objects. Exact observation fields, depth estimation, normalizations, and tensor dimensions are given in Appendix B.

3.3 SHARED RELATIONAL GRAPH ENCODER

Visual discrepancy is observed locally at each object, whereas the consequence of transforming that object depends on its relations to the rest of the scene. We therefore first ground each object in its visual and structured state, and then contextualize it through pairwise relations. The resulting node and relation features are shared by the actor and critic.

Object grounding. Omitting the step index t, we encode the visual evidence and fuse it with the structured object state:

$$z_i^{\text{vis}} = f_{\text{vis}}(V_i), \qquad h_i = f_{\text{node}}([x_i, z_i^{\text{vis}}]). \tag{4}$$

This fusion lets the policy interpret an image discrepancy jointly with the object's geometric and physical condition.

Relational contextualization. We condition each directed relation on its two endpoints and propagate relation context back to the objects:

$$\begin{aligned} r_{ij}^{(0)} &= f_{\text{edge}}([e_{ij}, h_i, h_j]), \\ m_i &= \text{Mean}_{j \in \mathcal{N}_i} f_{\text{msg}}(r_{ij}^{(0)}), \qquad \widetilde{h}_i = f_{\text{ctx}}([h_i, m_i]), \\ r_{ij}^{(1)} &= f_{\text{edge}}([e_{ij}, \widetilde{h}_i, \widetilde{h}_j]). \end{aligned} \tag{5}$$

The reverse direction is obtained by swapping the endpoints and the corresponding directional relation features. Thus, $\widetilde{h}_i$ supports object-centered decisions with incident relational context, while $r_{ij}^{(1)}$ supports pair-conditioned decisions whose endpoints already reflect the surrounding scene. Network widths, padding, and directional feature conventions are implementation details deferred to the appendix.

3.4 STRUCTURED ACTOR AND SCENE-LEVEL CRITIC

The shared graph representation serves two different roles. Action selection must preserve which object or relation an action operates on, whereas value estimation should summarize the scene as a whole. We therefore retain entity identity until policy scoring, while pooling the same encoded features for the critic.

Node policy heads operate on $\widetilde{h}_i$ for object-centered actions, and directed relation heads operate on $r_{ij}^{(1)}$ for pair-conditioned actions. The policy predicts a discrete operator k together with a continuous magnitude latent u,

$$\pi_\theta(k, u \mid \mathcal{G}) = p_\theta(k \mid \mathcal{G})\, \mathcal{N}(u; \mu_{\theta,k}(\mathcal{G}), \sigma^2), \tag{6}$$

forming a parameterized action space (Hausknecht & Stone, 2016). The discrete choice identifies the transformation and its object or endpoint, while u controls its bounded magnitude. Invalid and padded actions are masked.

For the critic, we pool contextualized node features together with symmetrized pair features:

$$\mathsf{V}_\phi(\mathcal{G}) = f_{\text{value}}\Big([\text{Pool}_{\mathcal{V}}(\{\widetilde{h}_i\}), \text{Pool}_{\mathcal{E}}(\{\tfrac{1}{2}(r_{ij}^{(1)} + r_{ji}^{(1)})\})]\Big). \tag{7}$$

The actor and critic share the visual, object, and relational encoders but retain separate task-specific heads. The critic estimates future return; physical acceptance remains the responsibility of the external verifier.

3.5 EXECUTABLE VERIFICATION AND VERIFIABLE REWARDS

After each action, the verifier recomputes consequences directly from rendering and simulation. We separate its task signal into two interpretable components: a *physical reward* that favors physically admissible layouts, and a *visual reward* that preserves consistency with the observation. A weak layout prior discourages unnecessary displacement from the initial reconstruction when several layouts satisfy the task.

Physical verification. Physical consistency is evaluated from pair penetration, support gap, excessive tilt, and residual motion under simulation. Let the corresponding non-negative residuals be L_{pair}, L_{sup}, L_{tilt}, and L_{roll}. We define the soft physical cost

$$C_{\text{phy}}(\mathcal{L}) = \sum_{v \in \mathcal{K}_{\text{phy}}} w_v \log(1 + L_v(\mathcal{L})), \qquad \mathcal{K}_{\text{phy}} = \{\text{pair}, \text{sup}, \text{tilt}, \text{roll}\}. \tag{8}$$

Static residuals expose immediate geometric violations, while simulated motion captures dynamic instability that static checks may miss. Exact residual definitions and normalization scales are given in Appendix B.

The verifier also defines a hard physical acceptance predicate. Let p_{ij} be pair penetration, g_i the signed support gap, α_i the tilt angle, and D and Ω the residual translation and rotation measured during simulation. A layout is physically accepted when

$$F(\mathcal{L}) = [\max_i |g_i| \le \eta_g] \wedge [\max_{i<j} p_{ij} \le \eta_p] \wedge [\max_i \alpha_i \le \eta_\alpha] \wedge [D \le \eta_D] \wedge [\Omega \le \eta_\Omega]. \tag{9}$$

Visual verification. Physics alone could favor a stable layout that drifts away from the input observation. Let $I(\mathcal{L})$ denote object-averaged mask IoU, $I_0 = I(\mathcal{L}_0)$, and E_i the normalized depth discrepancy. We use

$$\begin{aligned} L_{\text{vis}}(\mathcal{L}) &= 1 - I(\mathcal{L}) + \frac{\beta_{\text{d}}}{N} \sum_i E_i, \\ L_{\text{bar}}(\mathcal{L}) &= [I_0 - I(\mathcal{L})]_+, \\ C_{\text{vis}}(\mathcal{L}) &= w_{\text{vis}} \log(1 + L_{\text{vis}}) + w_{\text{bar}} \log(1 + L_{\text{bar}}). \end{aligned} \tag{10}$$

The first term provides dense image-space feedback, while the barrier explicitly penalizes degradation below the initial reconstruction.

Verifiable rewards. The two task rewards measure verified progress between consecutive layouts:

$$r_{\text{phy},t} = C_{\text{phy}}(\mathcal{L}_t) - C_{\text{phy}}(\mathcal{L}_{t+1}), \qquad r_{\text{vis},t} = C_{\text{vis}}(\mathcal{L}_t) - C_{\text{vis}}(\mathcal{L}_{t+1}). \tag{11}$$

We additionally use $C_{\text{prior}}(\mathcal{L}) = w_{\text{prior}} \log(1 + L_{\text{prior}}(\mathcal{L}))$ and $r_{\text{prior},t} = C_{\text{prior}}(\mathcal{L}_t) - C_{\text{prior}}(\mathcal{L}_{t+1})$ as a weak regularizer. This keeps the physical and visual rewards semantically distinct rather than assigning the pose prior to either objective.

Joint success requires both physical acceptance and preservation of the initial mask agreement,

$$S(\mathcal{L}) = F(\mathcal{L}) \wedge [I(\mathcal{L}) \ge I_0]. \tag{12}$$

The learning reward is

$$r_t = r_{\text{phy},t} + r_{\text{vis},t} + r_{\text{prior},t} - c_{\text{step}} + b_{\text{succ}} \mathbb{I}[S(\mathcal{L}_{t+1})]. \tag{13}$$

The dense physical and visual terms provide credit before all criteria are simultaneously satisfied, while the success bonus rewards reaching a physically accepted layout without sacrificing the initial visual agreement. We train the shared actor–critic with PPO (Schulman et al., 2017) and GAE (Schulman et al., 2016); Eq. (13) is the sole task-level supervision. At inference, we choose the highest-probability discrete action and the conditional mean of its continuous parameter.

4 Experiments

4.1 Experimental settings

Implementation Details. For physics simulation, we use MuJoCo (Todorov et al., 2012) and employ CoACD (Wei et al., 2022) to extract the collision geometry for each reconstructed object. We train PAIR3D from scratch using pPPO (Schulman et al., 2017) for 20,000 environment timesteps on all 1,600 training scenes, with random three seeds. Each PPO update collects 128 transitions and performs four optimization epochs with a minibatch size of 32. We use Adam (Kingma & Ba, 2015) with a learning rate of 2×10^{-4}, discount factor $\gamma = 0.97$, GAE parameter $\lambda = 0.95$, clipping threshold 0.2, value-loss coefficient 0.5, and entropy coefficient 0.01. Gradients are clipped to a maximum norm of 0.5. The policy receives node features, edge features, and four-channel visual tensors per object. The visual encoder produces 64-dimensional embeddings, and the graph policy uses a hidden dimension of 128. Episodes begin from the original SAM3D reconstruction and last at most 24 actions, terminating earlier upon success or a STOP action.

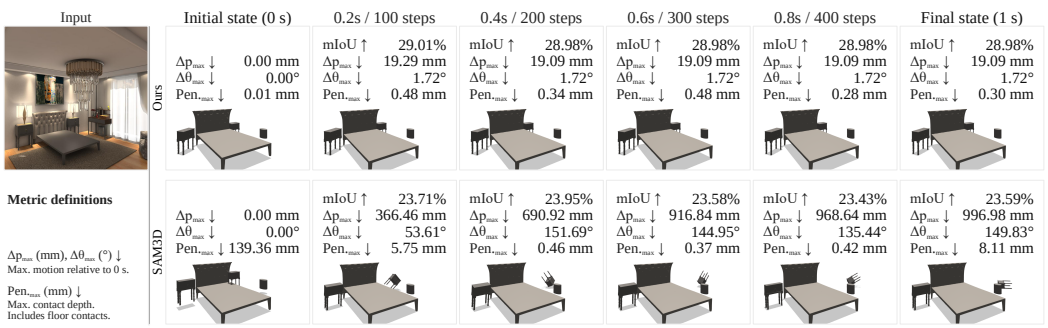


Figure 3: Evolution of reconstructed scenes under physics simulation. Starting from the predicted scene layouts, we simulate both reconstructions under the same physical conditions. PAIR3D largely preserves its initial configuration, whereas objects reconstructed by SAM3D undergo substantial translation and rotation, indicating greater physical stability of the scene reconstructed by PAIR3D.

Baselines. We compared the method proposed in this paper with three state-of-the-art (SOTA) methods for single-image scene reconstruction: SceneGen (Meng et al., 2026), SAM3D (Chen et al., 2026), and REST3D (Ma et al., 2026). All methods were evaluated using their publicly available code and pre-trained models. Additionally, we compared our approach against OpenAI's recently released agent, GPT-6 Astra (OpenAI, 2026), to assess the capabilities of its powerful 3D modeling features regarding visual and physical consistency.

Datasets. We construct 3D-F2Scene from SAM3D reconstructions of 1,800 synthetic scenes rendered using 3D-FUTURE (Fu et al., 2021a) furniture assets and 3D-FRONT (Fu et al., 2021b) layouts. SAM3D takes a 512 $\times$ 512 RGB image together with instance masks as input, and its reconstructed meshes, scales, and poses define the initial physical state of each scene. The dataset retains naturally occurring reconstruction errors, such as floating objects, inter-object penetration, and object tilt. Each scene includes input instance masks, MoGe-estimated depth and camera parameters, coordinate transformations, and a corresponding MuJoCo scene. We place a shared floor plane at the height of the lowest reconstructed vertex in each scene, thereby preserving relative vertical inconsistencies among objects. We employ a fixed scene-level split, consisting of 1,600 training scenes, 100 test scenes, and the remaining 100 scenes for validation and ablation studies.

Evaluation Metrics. For geometric metrics, mIoU measures the overlap between rendered visible masks and target masks. Chamfer Distance (CD) measures the bidirectional nearest-neighbor distance between reconstructed and ground-truth point set. F-Score combines precision and recall at a 0.05 m matching threshold. For physical metrics, Support measures absolute object–ground gaps, Penetration measures pairwise penetration depths, and Tilt measures deviations from upright. Rollout translation (RT) and rollout rotation (RR) measure object motion during physics simulation. We report scene-level pass rates using maximum-error thresholds of 0.005 m for Support, Penetration, and RT, and 5° for Tilt and RR. Physical-five measures the proportion of scenes satisfying all five criteria. For efficiency metrics, inference time (IT) measures average repair inference time in seconds per scene, and simulation calls (SC) measures the average number of complete simulation rollouts per scene.

4.2 Evaluating Scene Stability with Physics Simulation

We evaluate physical stability by rolling out each reconstructed scene in a physics simulator and tracking object poses over one second. As shown in Figure 3, PAIR3D remains nearly unchanged throughout the simulation, with a maximum object translation of only 19.1 mm and a maximum rotation of 1.7°. In contrast, SAM3D exhibits substantial motion after simulation, reaching 997.0 mm of translation and 149.8° of rotation. This suggests that a visually plausible initial reconstruction does not necessarily correspond to a physically stable scene. Importantly, PAIR3D achieves this stability while remaining close to its initial scene configuration. The relative object layout inferred from the input image is largely preserved throughout the rollout. The example therefore illustrates that

Table 1: Comparison of geometric, physical, and efficiency metrics.

Method	Geometric Metrics			Physical Metrics						Efficiency	
	mIoU (%) ↑	CD (mm) ↓	F-Score (%) ↑	Sup. (%) ↑	Penetr. (%) ↑	Tilt (%) ↑	RT (%) ↑	RR (%) ↑	Physical-five (%) ↑	IT (s) ↓	SC ↓
SceneGen	25.83	325.38	9.92	0.0	90.0	7.0	1.0	16.0	0.0	—	—
SAM3D	**46.56**	296.69	12.75	9.0	87.0	81.0	30.0	88.0	5.0	—	—
REST3D	45.48	295.47	**12.93**	50.0	**100.0**	**99.0**	**95.0**	**100.0**	48.0	**43.16**	63264.72
GPT-6	44.61	**295.26**	12.71	**68.0**	**93.0**	95.0	76.0	**93.0**	**62.0**	160.75	**4.14**
PAIR3D	**47.25**	**292.15**	**12.99**	**100.0**	**100.0**	**100.0**	**97.0**	**100.0**	**97.0**	**4.57**	**24.90**

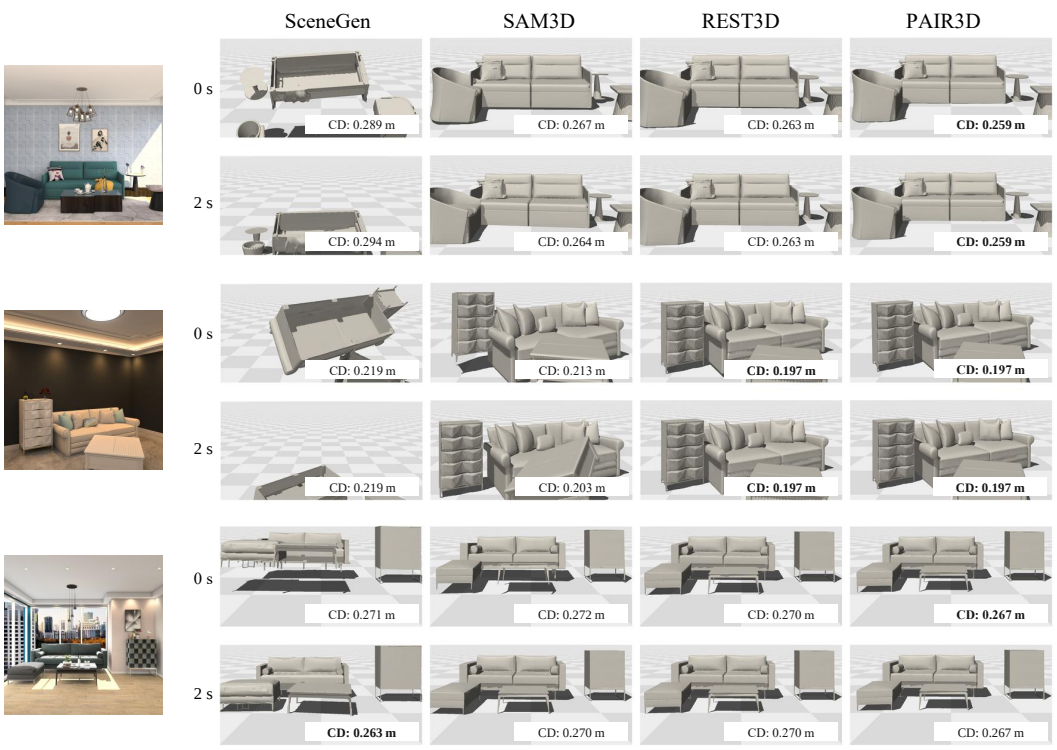


Figure 4: Comparison of physical simulations for scenes reconstructed using different methods. We visualize the reconstructed scenes in the MuJoCo physics simulator at the initial state (timestep = 0) and after 2 seconds of simulation (timestep = 1000). We present the results for the geometric metric CD in the bottom-right corner of each image.

PAIR3D improves physical consistency while preserving the image-supported scene arrangement, aligning the reconstructed scene more closely with both visual evidence and physical constraints.

4.3 Quantitative Comparison

Table 1 compares geometric accuracy, physical validity, and inference efficiency. PAIR3D achieves the best performance on all three geometric metrics, with 47.25% mIoU, 292.15 mm CD, and 12.99% F-Score. However, the margins over the strongest baseline on each metric are modest. This indicates that the primary advantage of PAIR3D lies not in improved geometric reconstruction, but in improving physical validity while preserving geometric accuracy.

The improvement is much larger under physical evaluation. PAIR3D achieves 100% pass rates on Support, Penetration, Tilt, and RR, and 97% on RT, resulting in a 97% Physical-five pass rate. The strongest competing result is 62% from GPT-6, leaving a 35-point gap on this scene-level metric. The per-criterion results also reveal why strong individual scores do not necessarily translate into physically valid scenes: REST3D attains high scores on Penetration, Tilt, RT, and RR, but its 50% Support rate limits Physical-five to 48%; GPT-6 shows a similar discrepancy, with 68% Support and 62% Physical-five. In contrast, PAIR3D satisfies the Support criterion for all evaluated scenes, with the remaining Physical-five failures arising only from RT.

Table 2: Ablation study of PAIR3D . We report mean ± standard deviation over three training seeds.

Method	mIoU (%) ↑	CD (mm) ↓	Physical-five (%) ↑
PAIR3D	**44.49 ± 0.07**	**316.79 ± 0.42**	**98.33 ± 1.53**
w/o Scale Adjustment	43.57 ± 0.24	317.30 ± 0.47	98.00 ± 2.00
w/o Visual Reward	42.40 ± 0.10	319.05 ± 0.12	93.00 ± 6.24
w/o Vision Branch	43.35 ± 0.31	317.11 ± 0.65	98.33 ± 0.58
w/o Relational Branch	43.44 ± 0.34	316.93 ± 0.52	97.00 ± 1.00

These gains are achieved with low inference cost. PAIR3D requires 4.57 s per scene, compared with 43.16 s for REST3D and 160.75 s for GPT-6, corresponding to $9.4\times$ and $35.2\times$ lower inference time, respectively. It uses 24.90 simulation rollouts per scene, substantially fewer than REST3D. Although GPT-6 uses fewer simulator calls (4.14), its end-to-end runtime is considerably higher. Thus, PAIR3D achieves substantially higher scene-level physical validity while retaining geometric accuracy and the lowest reported inference time.

4.4 Qualitative comparison

Figure 4 compares the simulation behavior of reconstructed scenes across different methods. At 0 s, SceneGen exhibits noticeable orientation and placement errors in several scenes. SAM3D more faithfully reproduces the input layout, but its reconstructions often suffer from substantial inter-object penetration and inaccurate floor contact. After 2 s of MuJoCo simulation, these differences become more pronounced. Both SceneGen and SAM3D exhibit substantial object translation or rotation, resulting in tilted furniture and displaced objects that disrupt the reconstructed layout and reduce its correspondence with the input image. In contrast, PAIR3D undergoes substantially less motion, better preserving object orientations, relative positions, and the overall projected scene structure. Compared with REST3D, PAIR3D further exhibits more stable physical grounding. Although neither method shows detectable initial inter-object penetration under the collision-based metric, PAIR3D achieves smaller object displacement across all scenes after 2 s of simulation. These results demonstrate that PAIR3D achieves a better balance between visual consistency and physical plausibility.

4.5 Ablation Studies

Table 2 examines the main design choices of PAIR3D . Restricting the action space to pose-only adjustment slightly degrades mIoU and CD while leaving physical success largely unchanged, suggesting that scale mainly provides additional flexibility for finer alignment. Removing the visual reward causes the largest degradation, reducing mIoU from 44.49% to 42.40% and Physical-five from 98.33% to 93.00%, highlighting the importance of jointly preserving visual consistency and physical plausibility.

Removing the vision encoder mainly reduces mIoU while leaving the already saturated physical metrics largely unchanged. This suggests that dense spatial evidence primarily supports finer observation-consistent refinement. Removing the pairwise relation encoder reduces mIoU to 43.44% and Physical-five to 97.00%, supporting the role of relational reasoning in capturing inter-object physical constraints.

5 Conclusion

We introduced PAIR3D, a reinforcement-learning framework for physics-aligned and visually consistent refinement of 3D scene layouts. PAIR3Dlearns a reusable policy from simulator-verifiable rewards, eliminating the need for expensive per-scene optimization. By jointly reasoning over visual evidence, object states, and pairwise relations, the policy efficiently improves physical validity while preserving consistency with the input image. Experiments show substantial gains in physical plausibility and reductions in inference time and simulator evaluations. The method currently refines object layouts while keeping geometry fixed; extending physics alignment to joint layout and shape refinement is a promising direction.

AI USE STATEMENT

We used generative AI tools to assist with literature discovery, manuscript drafting and editing, feedback on experimental methodology, and interpretation of results. The authors are responsible for validating all AI-assisted contributions and for the final content, data, and results.

REFERENCES

Andreea Ardelean, Mert Özer, and Bernhard Egger. Generalizable 3D scene reconstruction via divide and conquer from a single view. In *International Conference on 3D Vision*, 2025.

Dominik Bauer, Timothy Patten, and Markus Vincze. ReAgent: Point cloud registration using imitation and reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14586–14594, 2021.

Dominik Bauer, Timothy Patten, and Markus Vincze. SporeAgent: Reinforced scene-level plausibility for object pose refinement. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 654–662, 2022.

Boyuan Chen, Hanxiao Jiang, Shaowei Liu, Saurabh Gupta, Yunzhu Li, Hao Zhao, and Shenlong Wang. PhysGen3D: Crafting a miniature interactive world from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6178–6189, 2025.

Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J. Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, Aohan Lin, Jia-Wei Liu, Ziqi Ma, Anushka Sagar, Bowen Song, Xiaodong Wang, Jianing Yang, Bowen Zhang, Piotr Dollár, Georgia Gkioxari, Matt Feiszli, and Jitendra Malik. SAM 3D: 3Dfy anything in images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7220–7232, 2026.

Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang style, et al. 3d-front: 3d houses ranked with objects. In *IEEE/CVF International Conference on Computer Vision*, 2021a.

Huan Fu, Rongfei Jia, Lin Gao, Mingli Song, Chao-Hui Shen style, et al. 3d-future: 3d furniture shaped with texture. *International Journal of Computer Vision*, 2021b.

Matthew Hausknecht and Peter Stone. Deep reinforcement learning in parameterized action space. In *International Conference on Learning Representations*, 2016.

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. In *Robotics: Science and Systems*, 2026.

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

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

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.

Ke Li and Jitendra Malik. Learning to optimize. In *International Conference on Learning Representations*, 2017.

Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. DSO: Aligning 3D generators with simulation feedback for physical soundness. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6772–6783, 2025.

Guying Lin, Kemeng Huang, Michael Liu, Ruihan Gao, Hanke Chen, Lyuhao Chen, Beijia Lu, Taku Komura, Yuan Liu, Jun-Yan Zhu, and Minchen Li. PAT3D: Physics-augmented text-to-3D scene generation. In *International Conference on Learning Representations*, 2026.

Xiaoxuan Ma, Jiashun Wang, Nicolás Ugrinovic, Yehonathan Litman, and Kris Kitani. REST3D: Reconstructing physically stable 3D scenes from a single image. *SIGGRAPH Asia*, 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*, pp. 543–553, 2026.

Junfeng Ni, Yixin Chen, Bohan Jing, Nan Jiang, Bin Wang, Bo Dai, Puhao Li, Yixin Zhu, Song-Chun Zhu, and Siyuan Huang. PhyRecon: Physically plausible neural scene reconstruction. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

OpenAI. GPT-6 Astra: A new generation of intelligence. `https://openai.com/index/gpt-6-astra/`, 2026.

Tobias Sautter, Jan-Niklas Dihlmann, and Hendrik P. A. Lensch. 3D-RE-GEN: 3D reconstruction of indoor scenes with a generative framework. *arXiv preprint arXiv:2512.17459*, 2025.

John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations*, 2016.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

Xinyue Wei, Minghua Liu, Zhan Ling, and Hao Su. Approximate convex decomposition for 3d meshes with collision-aware concavity and tree search. *ACM Transactions on Graphics*, 2022.

Yandan Yang, Baoxiong Jia, Peiyuan Zhi, and Siyuan Huang. PhyScene: Physically interactable 3D scene synthesis for embodied AI. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16262–16272, 2024.

Kaixin Yao, Longwen Zhang, Xinhao Yan, Yan Zeng, Qixuan Zhang, Lan Xu, Wei Yang, Jiayuan Gu, and Jingyi Yu. CAST: Component-aligned 3D scene reconstruction from an RGB image. *ACM Transactions on Graphics*, 44(4), 2025.

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

Qingcheng Zhao, Xiang Zhang, Haiyang Xu, Zeyuan Chen, Jianwen Xie, Yuan Gao, and Zhuowen Tu. DepR: Depth guided single-view scene reconstruction with instance-level diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5722–5733, 2025.

A EXPERIMENTAL SETTINGS AND IMPLEMENTATION DETAILS

This appendix instantiates the symbolic formulation of PAIR3D with the configuration used in the existing experiments. Tables 3–4 map the representation and verifier symbols to their numerical values; Table 7 specifies training. Geometry, initial object scale, target masks, estimated target depth, camera, and floor plane are fixed during a policy episode.

A.1 NOTATION AND EXPERIMENTAL CONFIGURATION

The symbols in the main text separate the method definition from this experimental configuration. The numerical values below instantiate the same observation, action, reward, and optimization rules; they do not introduce additional model variants.

Symbol	Meaning	Experimental value
d_x, d_e	Node and relation descriptor dimensions	40, 28
C_v, H_v, W_v	Visual channels, height, width	4, 128, 128
d_v, d_h	Visual-embedding and hidden dimensions	64, 128
$N_{\max}$	Padded object capacity	8
$K_{\mathrm{n}}, K_{\mathrm{e}}$	Actions per object and unordered pair	18, 8
$\Delta_{\min}, \Delta_{\max}$	Compensation coordinate cutoff and norm cap	0.0005, 0.04 m
R_{pos}	Translation radius and pose-prior scale	0.55 m
$z_{\min}$	Minimum valid reference depth	10^{-6} m
$\chi_{\max}$	Maximum system condition number	10^8
$K_{\mathrm{warm}}, K_{\mathrm{meas}}$	Simulation warm-up and measurement steps	60, 60

Table 3: Representation, action, and verification-protocol settings for PAIR3D. Both compensation bounds are in meters.

Symbol	Meaning	Experimental value
β_{d}	Depth weight in visual discrepancy	0.25
s_p, s_g	Pair-penetration and floor-gap normalization scales	0.01, 0.02 m
s_α	Excess-tilt normalization scale	10°
s_D, s_Ω	Motion normalization scales	0.03 m, 5°
α_{tol}	Inactive range of the tilt penalty	2°
R_{rot}	Rotation-prior scale	π rad
η_g, η_p	Floor-gap and pair-penetration thresholds	0.0005 m each
η_α	Tilt acceptance threshold	2°
η_D	Translation-motion threshold	0.0015 m
η_Ω	Rotation-motion threshold	0.2°
$w_{\mathrm{vis}}, w_{\mathrm{bar}}$	Visual and visual-margin weights	2, 25
$w_{\mathrm{pair}}, w_{\mathrm{sup}}, w_{\mathrm{tilt}}$	Penetration, support, and tilt weights	8, 8, 4
$w_{\mathrm{roll}}, w_{\mathrm{prior}}$	Motion and pose-prior weights	2, 0.2

Table 4: Numerical instantiation of the verifier cost and acceptance predicates. Normalization scales and acceptance thresholds play distinct roles even when their values coincide.

A.2 OBSERVATION FIELDS AND RENDERING

Let $\mathbf{d}_i$ be the side lengths of the scaled local mesh's axis-aligned bounding box and $L = \max(1, \max_i \|\mathbf{d}_i\|_2)$. Write g_i for signed floor gap, p_i^{ground} for maximum ground penetration, p_{ij} for maximum object-pair penetration, I_i for mask IoU, and $\Delta\mathbf{t}_i^{\mathrm{roll}}, \Delta\boldsymbol{\omega}_i^{\mathrm{roll}}$ for measured simulation motion. The following indices are zero-based.

For reproducibility, the reverse-direction transform $\mathcal{S}$ negates indices 1–6, 14–16, and 19–22; swaps the extent blocks 7–9 and 10–12; and swaps the zero-valued slots 23 and 24. The existing implementation leaves the support-gap fields 26 and 27 in their original order. Accordingly, we do not claim exact permutation equivariance for this feature construction.

Indices	Node descriptor $x_i \in \mathbb{R}^{40}$
0	Constant one.
1–3	$(\mathbf{t}_i - [0, 0, z_{\text{floor}}]^\top)/L$.
4–9	First two columns of $\mathbf{R}_i$, flattened in row-major order.
10–12	$\mathbf{d}_i/L$.
13–16	Shape statistics: $\log(1 + n_v)/14$, $\log(1 + n_f)/14$, $\prod_k d_{ik}$, and $\max_k d_{ik} / \max(\min_k d_{ik}, 10^{-4})$.
17–20	$g_i/0.02$, $p_i^{\text{ground}}/0.01$, $\max_j p_{ij}/0.01$, and I_i. Distances are in meters.
21–24	Current minus target mask statistics $[\text{centroid}_x, \text{centroid}_y, x_{\max}, y_{\max}]/128$.
25–26	Normalized depth error E_i and $\|\Delta\mathbf{t}_i^{\text{roll}}\|_2/0.03$.
27–29	$\Delta\boldsymbol{\omega}_i^{\text{roll}}/\operatorname{rad}(5^\circ)$.
30–32	$\|\mathbf{t}_i - \mathbf{t}_i^0\|_2/0.55$, $\|\operatorname{Log}(\mathbf{R}_i(\mathbf{R}_i^0)^\top)\|_2/\pi$, and zero.
33–36	Camera features $[f_x^{\text{NDC}}/4, f_y^{\text{NDC}}/4, p_x^{\text{NDC}}, p_y^{\text{NDC}}]$.
37–39	Tilt angle divided by 180°, collision-label confidence, and initial scene mean IoU.

Table 5: Structured node observation. n_v, n_f are vertex and face counts; Log returns a rotation vector.

Indices	Relation descriptor $e_{ij} \in \mathbb{R}^{28}$
0	Constant one.
1–3	$(\mathbf{t}_j - \mathbf{t}_i)/L$.
4–6	$\operatorname{Log}(\mathbf{R}_j\mathbf{R}_i^\top)/\pi$.
7–12	$\mathbf{d}_i/L$ and $\mathbf{d}_j/L$.
13–16	$p_{ij}/0.01$ and the separation direction of the deepest contact.
17–18	Pairwise IoU of the current masks and of the target masks.
19–22	Current and target mask centroid differences, from i to j, divided by 128.
23–24	Two zero-valued reserved fields.
25–27	Collision-label confidence, $g_i/0.02$, and $g_j/0.02$.

Table 6: Structured relation observation. Masks are rendered independently per object.

Target masks and MoGe depth are resized to 128×128 by nearest-neighbor interpolation. Target depth is zero outside each target mask. For each object, the four channels are target mask, current mask, target depth, and current depth; both depth channels are divided by the median positive target depth, floored at 0.1 m, and clipped to $[0, 3]$. If no positive target depth exists, the normalization scale is 1. This input normalization uses all positive target depths, whereas the depth error below uses valid overlapping pixels.

The rasterizer uses one face per pixel, zero blur, no back-face culling, and the frozen perspective camera. Each mesh is rasterized independently, without inter-object occlusion. A stored visual proxy is used when available; otherwise meshes exceeding 12,000 faces are reduced by evenly spaced face selection. Graph tensors are padded to eight objects and 28 unordered relations, with separate node, edge, and action masks.

A.3 Network architecture

The vision encoder consists of three convolution–SiLU blocks: $(4, 16, 5, 2, 2)$, $(16, 32, 3, 2, 1)$, and $(32, 48, 3, 2, 1)$, where each tuple specifies input/output channels, kernel, stride, and padding. Global average pooling and a $48 \to 64$ linear–SiLU layer produce z_i^{vis}. The remaining encoders are two-layer MLPs with one intervening SiLU: node $104 \to 128 \to 128$, edge $284 \to 128 \to 128$, message $128 \to 128 \to 128$, and node update $256 \to 128 \to 128$. There is no additional activation after the final MLP layer.

Each node has separate linear heads for 18 logits and 18 Gaussian means. Each directed relation has corresponding heads for four endpoint translations, giving eight actions per unordered relation. The stop logit and mean are learned global scalars. The Gaussian log standard deviation $\log\sigma$ is shared across actions and initialized to -0.2. The critic is an MLP $256 \to 128 \to 1$ with SiLU between layers. The padded categorical distribution has $8 \times 18 + 28 \times 8 + 1 = 369$ entries; masked logits are set to -10^9.

Training begins from a fresh initialization. The 12 original rows in the node-logit and node-mean heads are copied from the freshly initialized base heads into expanded 18-row heads; six additional rows retain their fresh random initialization. No trained base-policy weights are used to initialize PAIR3D.

A.4 VERIFIER MEASUREMENTS AND REWARD COMPONENTS

Pair penetration is the deepest negative MuJoCo contact distance for each object pair, or zero without contact. Uprightness is $\alpha_i = \arccos(\mathbf{e}_z^\top \mathbf{R}_i \mathbf{e}_z)$. For motion measurement, the candidate poses and zero velocities initialize MuJoCo; after 60 warm-up steps, another 60 steps define the measurement window. Let

$$D = \max_i \|\Delta \mathbf{t}_i^{\mathrm{roll}}\|_2, \qquad \Omega = \max_i \|\Delta \boldsymbol{\omega}_i^{\mathrm{roll}}\|_2. \tag{14}$$

These are differences between the two measurement-window endpoints, not maximum intermediate excursion or total motion since initialization. Simulator states are used for measurement only. The exported simulation configuration uses a $0.002\,\mathrm{s}$ time step and RK4 integration, corresponding to $0.12\,\mathrm{s}$ for each phase.

Let $\mathcal{P}_i$ be pixels inside both target and current masks with positive depths in both. The normalized depth error is

$$E_i = \frac{\mathrm{mean}_{p\in\mathcal{P}_i} |D_i^\star(p) - D_i^t(p)|}{\max(\mathrm{median}_{p\in\mathcal{P}_i} D_i^\star(p), 0.1\,\mathrm{m})}, \tag{15}$$

with $E_i = 1$ if $\mathcal{P}_i$ is empty. Mask IoU is one for an empty union. Scene IoU is the unweighted mean over objects, $I = N^{-1}\sum_i I_i$.

The seven terms in reward function are

$$L_{\mathrm{vis}} = 1 - I + \frac{0.25}{N}\sum_i E_i, \tag{16}$$

$$L_{\mathrm{bar}} = \max(0, I_0 - I), \tag{17}$$

$$L_{\mathrm{pair}} = \kappa \frac{\sum_{i<j} p_{ij}}{0.01\,\mathrm{m}}, \tag{18}$$

$$L_{\mathrm{sup}} = \frac{\sum_i |g_i|}{0.02\,\mathrm{m}}, \tag{19}$$

$$L_{\mathrm{tilt}} = \frac{\sum_i \max(0, \alpha_i - 2^\circ)}{10^\circ}, \tag{20}$$

$$L_{\mathrm{roll}} = \frac{D}{0.03\,\mathrm{m}} + \frac{\Omega}{5^\circ}, \tag{21}$$

$$L_{\mathrm{prior}} = \sum_i \frac{\|\mathbf{t}_i - \mathbf{t}_i^0\|_2}{0.55\,\mathrm{m}} + \sum_i \frac{\|\log(\mathbf{R}_i(\mathbf{R}_i^0)^\top)\|_2}{\pi}. \tag{22}$$

Angles in a ratio use matching units; implementation rotation vectors are in radians. Collision confidence κ comes from the scene label, with $\kappa = 0.25$ for an unconfirmed coarse collision candidate. The weights in the displayed order are $(2, 25, 8, 8, 4, 2, 0.2)$. Floor penetration is penalized through the signed support-gap magnitude.

The physical acceptance predicate F requires all five thresholds:

$$\begin{aligned} &\max_i |g_i| \le 0.0005\,\mathrm{m}, \qquad \max_{i<j} p_{ij} \le 0.0005\,\mathrm{m}, \\ &\max_i \alpha_i \le 2^\circ, \qquad D \le 0.0015\,\mathrm{m}, \qquad \Omega \le 0.2^\circ. \end{aligned} \tag{23}$$

The training success predicate additionally requires $I \ge I_0$. A separate strict visual-preservation statistic requires F and $I \ge I_0$.

A.5 PPO AND INFERENCE PROTOCOL

One environment samples scenes uniformly from the fixed training split and resets each scene to its original reconstructed pose. Error-signature balancing is disabled. Python, NumPy, and PyTorch use

Parameter	Value
Environment transitions T_{train}	20,000
Maximum actions per episode $T_{\max}$	24
Rollout buffer B_{roll} / minibatch B_{mini}	128 / 32
PPO epochs per buffer K_{PPO}	4
Optimizer / learning rate η_{lr}	Adam / 2×10^{-4}
Discount γ / GAE λ	0.97 / 0.95
Policy clipping ϵ	0.2
Value coefficient c_V / entropy coefficient c_H	0.5 / 0.01
Gradient norm limit $g_{\max}$	0.5
Step cost c_{step} / success bonus b_{succ}	0.002 / 2
Historical training seed	23, 7816, 294683
Seeds in the PAIR3D ablation study	23, 7816, 294683

Table 7: Fixed experimental training settings for PAIR3D.

the specified run seed. The final rollout buffer contains 32 transitions and training stops at exactly 20,000 transitions.

Raw rewards are divided by a running standard deviation without centering or clipping. The online variance accumulator starts with count 10^{-4}, mean zero, and second-moment accumulator one; its variance is floored at 10^{-6}. Each reward is scaled after updating the accumulator with that reward. For the resulting r_t and terminal flag d_t,

$$\delta_t = r_t + \gamma(1 - d_t)\mathsf{V}_{\text{old}}(\mathcal{G}_{t+1}) - \mathsf{V}_{\text{old}}(\mathcal{G}_t), \tag{24}$$

$$A_t = \delta_t + \gamma\lambda(1 - d_t)A_{t+1}, \tag{25}$$

$$\widehat{R}_t = A_t + \mathsf{V}_{\text{old}}(\mathcal{G}_t). \tag{26}$$

Advantages are standardized within each collected buffer using denominator $\text{std}(A) + 10^{-8}$. Value targets use the unstandardized advantages. The joint log probability includes both the categorical choice and Gaussian latent, including the stop action's unused latent. Entropy is the sum of categorical and Gaussian entropy. There is no value clipping, auxiliary supervised loss, learning-rate schedule, or target-KL early stopping.

B Implementation record

Table 8: Dimensions confirmed in the supplied architecture description. d_m is the message width, which has not been independently specified.

Quantity or operation	Input or meaning	Output
Node attributes x_i	Structured object-state attributes	40
Visual tensor V_i	Current/target masks and depths	$4 \times 128 \times 128$
Pair descriptor e_{ij}	Spatial–physical attributes of a pair	28
Visual encoder	V_i	64
Node encoder	$[x_i, z_i^{\text{vis}}]$: $40 + 64 = 104$	128
Edge encoder	$[e_{ij}, h_i, h_j]$: $28 + 128 + 128 = 284$	128
Message transform	Relation embedding	d_m
Node update	$[h_i, m_i]$: $128 + d_m$	128
Actor	Updated nodes and retained relations	Policy outputs
Critic	Separate node and relation pooling	Scalar value