Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

Picasso: Holistic Scene Reconstruction with Physics-Constrained Sampling

Xihang Yu[†] Rajat Talak[‡] Lorenzo Shaikewitz[†] Luca Carlone[†]

[†] Massachusetts Institute of Technology, Cambridge, MA, 02139, USA
[‡] National University of Singapore, Singapore 117583
Emails: {jimmyyu, lorenzos, lcarlone}@mit.edu
Email: talak@nus.edu.sg

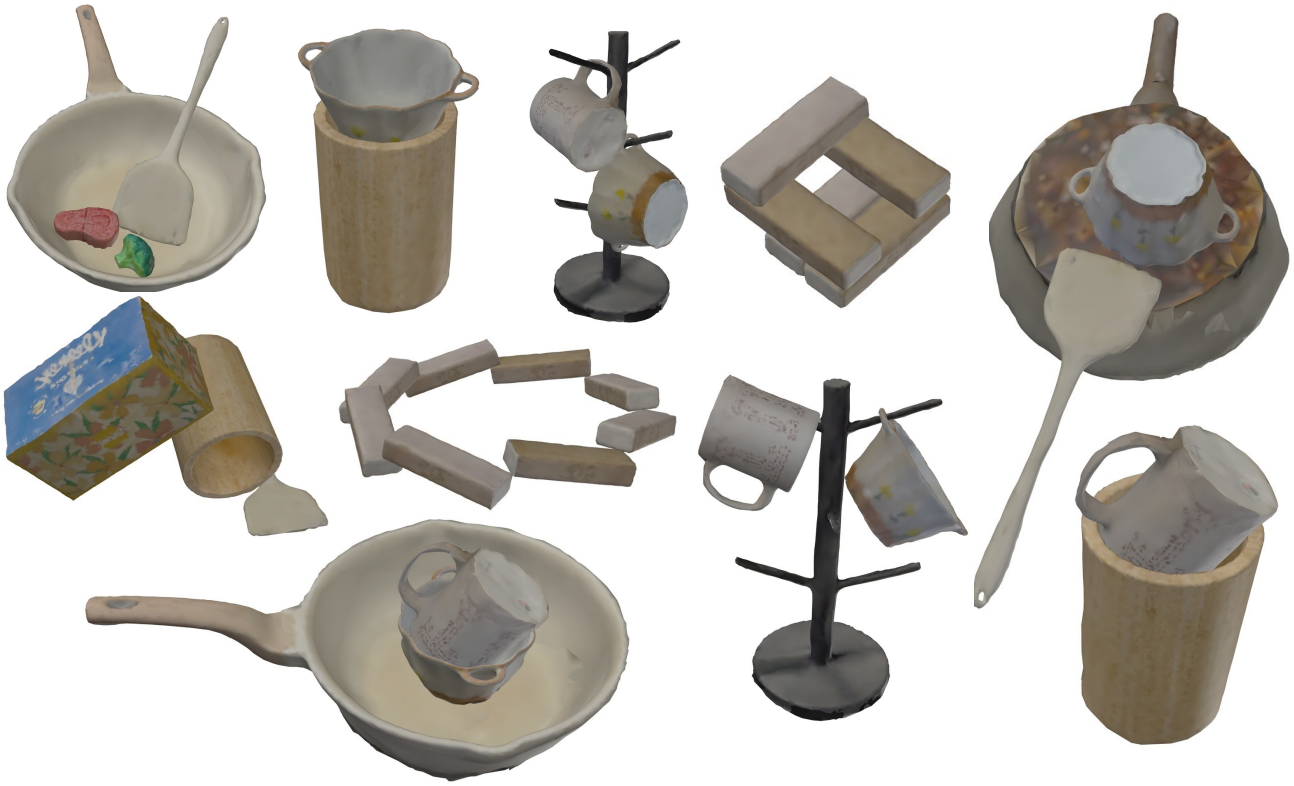


Fig. 1: We propose *Picasso*, an approach to build multi-object scene reconstructions by accounting for object geometry, non-penetration, and physics (*i.e.,* objects should be in a stable equilibrium for the scene to be static). We also release the *Picasso dataset*: a collection of 10 contact-rich real-world scenes we use to test physical plausibility of scene reconstructions. The figure shows the digital twins generated from the 10 real-world scenes using ground-truth pose annotations.

***Abstract*—In the presence of occlusions and measurement noise, geometrically accurate scene reconstructions—which fit the sensor data—can still be *physically incorrect*. For instance, when estimating the poses and shapes of objects in the scene and importing the resulting estimates into a simulator, small errors might translate to implausible configurations including object interpenetration or unstable equilibrium. This makes it difficult to predict the dynamic behavior of the scene using a digital twin, an important step in simulation-based planning and control of contact-rich behaviors. In this paper, we posit that object pose and shape estimation requires reasoning holistically over the scene (instead of reasoning about each object in isolation), accounting for object interactions and physical plausibility. Towards this goal, our first contribution is *Picasso*, a physics-constrained reconstruction pipeline that builds multi-object scene reconstructions by considering geometry, non-penetration, and physics. Picasso relies on a fast rejection sampling method that reasons over multi-object interactions, leveraging an inferred object contact graph to guide samples. Second, we propose the *Picasso dataset*, a collection of 10 contact-rich real-world scenes with ground truth annotations, as well as a metric to quantify physical plausibility, which we open-source as part of our benchmark. Finally, we provide an extensive evaluation of Picasso on our newly introduced dataset and on the YCB-V dataset, and show it largely outperforms the state of the art while providing reconstructions that are both physically plausible and more aligned with human intuition.**

This work was partially supported by Symbotic, the ONR RAPID program, Carlone's NSF CAREER award, and AFOSR "Certifiable and Self-Supervised Category-Level Tracking" Program.

L. Shaikewitz is supported by an NSF graduate research fellowship.

L. Carlone holds concurrent appointments as a faculty at the Massachusetts Institute of Technology and as an Amazon Scholar. This paper describes work performed at MIT and is not associated with Amazon.

I. Introduction

Imagine playing Jenga or building a toy castle with wooden blocks. To decide how to remove or add blocks, we would form a mental picture of the structure in front of us, and reason over relations between blocks (for instance, "block A supports blocks B and C"). This allows us to ponder different actions ("if I remove block A, two blocks will fall"); in other words, such a mental model serves as a digital twin (or, in modern terms, as an explicit *world model*) we can use to predict how the environment would respond to our actions. While being effortless for humans, the task of building such a world model is still challenging for robots, and requires forming a holistic scene understanding that accounts for both visual observations and physical plausibility. At the same time, unlocking this capability for our robots would enable contact-rich manipulation and reasoning in cluttered scenes, possibly beyond the reach of current vision-language-action models.

Despite the fast-paced progress in computer vision, graphics, and robotics, only a few works deal with physics-aware explicit world modeling [41]. Foundation models for 3D reconstruction [50] build a dense 3D reconstruction of a scene, but do not reason about interactions among objects. Work on object understanding, including recent foundation 3D vision models [9], most commonly focus on estimating pose and shape of objects, but process objects in isolation; this leads to artifacts where even fairly accurate pose estimates lead to physically implausible configurations that are immediately recognizable as incorrect by a human (Fig 2). To incorporate physics constraints, a common strategy is to design physics-guided losses, *e.g.,* discouraging copenetration and encouraging stable contact—and optimize them with gradient-based methods [2, 35, 60]. However, this approach is prone to converge to local minima, which are physically plausible but geometrically incorrect configurations. An alternative is to integrate a differentiable physics simulator to generate corrective signals (*e.g.,* [29, 36]); however, simulation-based supervision is often brittle in practice due to modeling inaccuracies and numerical noise in the simulation engine [60].

In this paper, we posit that object pose and shape estimation requires reasoning *holistically* over the scene (instead of reasoning about each object in isolation) and accounting for object interactions and physical plausibility. Towards this goal, we propose a model-based approach that retrofits existing object pose and shape estimators (*e.g.,* SAM3D [9]) to make their estimates more physically plausible.

In particular, we propose *Picasso* (Section III), a physics-constrained reconstruction pipeline that builds multi-object scene reconstructions by accounting for geometry, non-penetration, and contact. Instead of directly optimizing a physics-based loss, Picasso relies on fast rejection sampling. This has two advantages. First, by treating physics as constraints rather than as additional penalty terms, we avoid reshaping the objective with competing losses that may introduce extra local minima. Second, rejection sampling encourages global exploration of the state space, which further reduces the risk of being stuck in local minima. To make the sampling tractable, we sample poses that respect which objects are in contact. This reduces the complexity of reasoning over hypothetical interactions and reduces the dimensionality of the sampling space.

Fig. 2: An example illustrating that a 3D scene reconstruction from SAM3D [9] can be physically implausible. **Left**: The original image. **Right**: The reconstruction exhibits multiple penetrations, highlighted in the red boxes.

Our second contribution is the *Picasso dataset* (Section IV), a collection of 10 contact-rich real-world scenes with ground truth annotations, as well as novel metrics to quantify physical plausibility, which we open-source with our benchmark. The Picasso dataset is visualized in Fig. 1 (the figure shows the ground-truth digital twin of each scene, while sample images are given in Appendix D) and includes multiple contact-rich scenes found in domestic environments (*e.g.,* a pile of plates and pans in a sink, or a set of Jenga blocks).

Our final contribution is an *extensive evaluation* of Picasso (Section V), on both our newly introduced dataset and on the YCB-V dataset [56]. We show that the proposed approach (i) can easily retrofit modern object pose and shape estimators, (ii) outperforms state-of-the-art methods in terms of pose accuracy and physical plausibility, and (iii) the resulting reconstructions are more aligned with human intuition. Our code and dataset are available online[1].

II. Related Work

Object Pose and Shape Estimation. Object pose and shape estimation involves recovering the pose and shape of an object, typically from RGB or RGB-D observations. Existing methods can be classified into three categories: *instance-level*, *category-level*, and *category-agnostic*. Instance-level approaches assume the object shape is known and focus on pose estimation [56, 47, 45, 52]. In contrast, category-level methods estimate pose and shape of objects within the same object category (*e.g.,* cars), without requiring instance-specific CAD models. These approaches often learn to model shape deformations or normalized coordinate representations to capture intra-class variations [40, 49, 48, 18, 46, 61]. More recently, category-agnostic approaches have attracted increasing attention [25, 26, 1, 24]. Compared to category-level methods, which often rely on curated category annotations,

[1] https://github.com/MIT-SPARK/Picasso

category-agnostic approaches estimate object pose and shape without assuming knowledge of the object category, leading to more scalable and broadly applicable methods [12]. Most approaches in this area (*e.g.,* [51]) rely on multiple reference images to reconstruct object shape. In contrast, recent work [9] demonstrates single-image shape and pose estimation.

Physics-informed Scene Understanding. Existing methods to inject physical reasoning into scene understanding can be broadly classified into three classes. First, optimization-based methods impose physics-inspired constraints and solve for physically plausible states. Zhang et al. [62] focus on multi-object pose estimation by enforcing hard non-penetration constraints and solving the resulting problems using semi-infinite programming. Methods like PhysPose [2, 35, 60] instead define differentiable losses based on physics constraints like gravity and non-penetration and optimize the losses with gradient descent. Vysics and follow-up works [5, 65] assume a rigid contact model and solve for contact parameters, which are then used for shape estimation. Second, simulation-based approaches rollout dynamics to optimize physics parameters, see PhyRecon [36], TwinTrack [59], HoloScene [55] and PhysTwin [29]. Third, sampling approaches formulate physical plausibility in probabilistic terms and perform inference via sampling. Early work [63] measures plausibility through changes in system energy and optimizes for stability with Markov Chain Monte Carlo (MCMC). Another work, 3DP3 [19], models flush contact between objects (*i.e.,* forces two contact faces to be coplanar with zero offset) and uses MCMC to jointly infer object shapes, poses, and a scene graph. Follow-up work [20] uses Bayesian inference for objects in flush contact. Our approach is sampling-based but can handle arbitrary contact types (Section III-B) and accounts for co-penetration.

III. Picasso

Problem Formulation and Approach. Consider a scene with N objects. We are given an RGB image $\boldsymbol{I}$, depth map $\boldsymbol{D}$, along with object masks $\boldsymbol{M}_i, i \in [N]$. For each object $i \in [N]$ in the scene, our goal is to estimate its scale, translation, and rotation represented in the camera frame, namely $\boldsymbol{T}_i = (s_i, \boldsymbol{R}_i, \boldsymbol{t}_i) \in \mathrm{SIM}(3)$, as well as the object shape $\boldsymbol{S}_i$. We formulate this as maximum likelihood estimation (MLE):

$$\underset{\{\boldsymbol{T}_i, \boldsymbol{S}_i\}_{i=1}^N \in \mathcal{F}}{\arg\max} \; p\big(\{\boldsymbol{T}_i \,,\, \boldsymbol{S}_i\}_{i=1}^N \mid \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big)\,, \tag{1}$$

where $\mathcal{F}$ is the set of physically plausible pose and shape estimates. Problem (1) finds the most likely object configuration given the measurements, while satisfying physical plausibility.

Next, we derive expressions for the measurement likelihood (Section III-A) and constraints (Section III-B) in eq. (1), leading to the optimization problem in (Section III-C) eq. (12). Then we show how to approximate the problem as a set of lower-dimensional subproblems (one per object), using the notion of *contact scene graph* (Section III-D). Finally, we solve each subproblem via rejection sampling (Section III-E). Figure 3 shows a block diagram of the proposed approach.

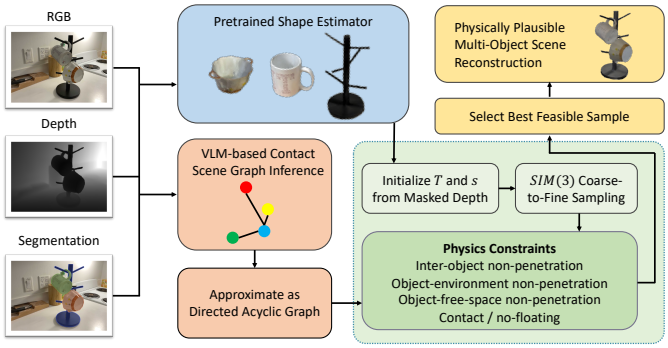


Fig. 3: System architecture of Picasso. A VLM is queried to infer the contact scene graph, and a pretrained shape estimator provides shape priors. A physics-constrained sampling solver then optimizes object poses by reasoning over multi-object relationships encoded in the contact graph.

A. Measurement Likelihood

We begin by defining the measurement likelihood objective in eq. (1). Observe that we can factor the probability in the objective of eq. (1) as:

$$\begin{aligned} p\big(\{\boldsymbol{T}_i \,,\, \boldsymbol{S}_i\}_{i=1}^N \mid \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) = \\ p\big(\{\boldsymbol{T}_i\}_{i=1}^N \mid \{\boldsymbol{S}_i\}_{i=1}^N, \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) \times \\ p\big(\{\boldsymbol{S}_i\}_{i=1}^N \mid \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) \end{aligned} \tag{2}$$

On the left side of (2), the first term captures pose estimation given the shape, and the second term is shape estimation.

Shape Likelihood. Instead of analytically expanding the shape likelihood, we assume the availability of a network, *e.g.,* [9], that performs shape prediction given sensor data. We can deal with both the case in which the network returns a single shape estimate as in [9], as well as the case where the network returns a number of potential estimates (*e.g.,* by retrieving the k most similar shapes in the training set [46]), in which case the probability $p\big(\{\boldsymbol{S}_i\}_{i=1}^N \mid \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big)$ becomes a uniform distribution over discrete options. For simplicity, in the rest of this section we assume the network provides a single shape estimate and focus on pose estimation.

Pose Likelihood. In contrast to the shape, we will expand the pose likelihood $p\big(\{\boldsymbol{T}_i\}_{i=1}^N \mid \{\boldsymbol{S}_i\}_{i=1}^N, \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big)$ analytically. Under the standard assumption of independent measurement noise, the probability factors as:

$$\begin{aligned} p\big(\{\boldsymbol{T}_i\}_{i=1}^N \mid \{\boldsymbol{S}_i\}_{i=1}^N, \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) = \\ \prod_{i=1}^N p(\boldsymbol{T}_i \mid \boldsymbol{S}_i, \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{M}_i) \end{aligned} \tag{3}$$

where we also used the fact that an object's pose is independent from the masks and shapes of other objects. Note that this does not mean the poses can be estimated independently; the resulting estimates are still coupled by the constraints in (1).

Now $p(\boldsymbol{T}_i \mid \boldsymbol{S}_i, \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{M}_i)$ is simply the likelihood of an object pose given its shape and measurements. Using RGB-D measurements, this can be framed as a point cloud registration problem. In particular, if we denote as $\mathcal{A}_i = \{\boldsymbol{a}_{i,j}\}_{j=1}^{M_{\mathcal{A}_i}}$ the set of 3D points measured by the RGB-D camera which lie in

the mask $\boldsymbol{M}_i$ of object i, we can write the likelihood of the measurements as:

$$p(\boldsymbol{T}_i \mid \boldsymbol{S}_i, \boldsymbol{I}, \boldsymbol{D}, \boldsymbol{M}_i) \propto \exp\left(-\sum_{j=1}^{M_{\mathcal{A}_i}} d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \boldsymbol{S}_i)\right) \tag{4}$$

where the distribution is chosen to be in the exponential family (most commonly a Gaussian, if the noise is assumed to be additive and normally distributed), and depends on the distance $d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \boldsymbol{S}_i)$ of the measured point $\boldsymbol{a}_{i,j}$ (transformed to the object frame via $s_i, \boldsymbol{R}_i, \boldsymbol{t}_i$) to the surface of the object (as described by its shape $\boldsymbol{S}_i$).

Substituting (4) back into (3) and observing that maximizing the likelihood is the same as minimizing the negative log-likelihood leads to:

$$\begin{aligned}
&\underset{\{\boldsymbol{T}_i\}_{i=1}^N \in \mathcal{F}}{\arg\max} \prod_{i=1}^{N} p\big(\{\boldsymbol{T}_i\}_{i=1}^N \mid \{\boldsymbol{S}_i\}_{i=1}^N, \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) \\
&= \underset{\{\boldsymbol{T}_i\}_{i=1}^N \in \mathcal{F}}{\arg\min} \; -\log \prod_{i=1}^{N} p\big(\{\boldsymbol{T}_i\}_{i=1}^N \mid \{\boldsymbol{S}_i\}_{i=1}^N, \boldsymbol{I}, \boldsymbol{D}, \{\boldsymbol{M}_i\}_{i=1}^N\big) \\
&= \underset{\{\boldsymbol{T}_i\}_{i=1}^N \in \mathcal{F}}{\arg\min} \sum_{i=1}^{N} \sum_{j=1}^{M_{\mathcal{A}_i}} d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \boldsymbol{S}_i)\,.
\end{aligned} \tag{5}$$

While the expression of the distance in (5) depends on the shape representation (*e.g.,* SDF, mesh), a common approach is to sample a set of points $\mathcal{B}_i = \{\boldsymbol{b}_{i,k}\}_{k=1}^{M_{\mathcal{B}_i}}$ from the object's surface and then use a point-to-point distance as metric, leading to the well-known Chamfer distance [46]:

$$\begin{aligned}
d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \boldsymbol{S}_i) &\doteq d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \mathcal{B}_i) \\
&= \textstyle\sum_{j=1}^{M_{\mathcal{A}_i}} \min_{1 \le k \le M_{\mathcal{B}_i}} \|s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i - \boldsymbol{b}_{i,k}\|^2
\end{aligned} \tag{6}$$

where, for each measured point $\boldsymbol{a}_{i,j}$, the objective computes the distance to the closest point in the object point cloud $\mathcal{B}_i$.

Remark 1 (The Single-Object Case)**.** *When $N = 1$, (5) reduces to standard point-cloud registration: estimate the transform aligning two point clouds. A common solution strategy is to alternate between finding the correspondences between points in the two point clouds and computing the best transformation given the correspondences. For instance, the celebrated Iterative Closest Point (ICP) follows this paradigm. However, this line of approaches is prone to finding local minima. Other correspondence-free paradigms include optimization-based and sampling-based methods. Optimization-based methods phrase the problem as a mixed-integer program (MIP) and either solve it using MIP solvers [27], Branch-and-Bound (BnB) [7, 23, 38, 58, 20], or convex relaxations [57]. While these are effective with strong correspondence priors settings [57, 33], they fail in the correspondence-free case [57]. Alternatively, sampling-based methods sample potential pose hypothesis until one that achieves a sufficiently small objective is found [58]. These approaches have a close relationship with MCMC [19] and render-and-compare approaches [32, 52], which render* $\mathrm{SE}(3)$ *transformation hypotheses and compare the rendered objects at those poses against the observed RGB or depth maps.*

Takeaway

In the single object case, sampling is an effective means to tackle problem (5) since (i) it only requires evaluating the highly nonconvex objective in (5), (ii) we can sample from a relatively low-dimensional space ($\mathrm{SIM}(3)$), and (iii) evaluating each sample is trivially parallelizable.

B. Physics Constraints

In this section, we derive the physics-based constraints used in eq. (1). We focus on *static* scenes and restrict ourselves to configuration-space (holonomic) constraints, namely non-penetration and contact proximity. We assume the objects to lay on a planar surface (the *environment*, *e.g.,* a table or ground floor), whose parameters we estimate beforehand (using a standard 3-point RANSAC from the depth measurements [16]).

Let $\mathcal{B}_i = \{\boldsymbol{b}_{i,k}\}_{k=1}^{M_{\mathcal{B}_i}} \subset \mathbb{R}^3$ denote vertices (or sampled surface points) in the local frame of object i and recall that $\boldsymbol{T}_i \in \mathrm{SIM}(3)$ maps points in the camera frame to the object frame. Moreover, let $\Phi_i : \mathbb{R}^3 \to \mathbb{R}$ be the signed distance field (SDF) of object i in the local frame. Similarly, let $\Phi_{\text{env}} : \mathbb{R}^3 \to \mathbb{R}$ denote the environmental SDF and $\Phi_{\text{free}} : \mathbb{R}^3 \to \mathbb{R}$ denote the free-space SDF (describing the geometry of the free space) respectively. We enforce physical plausibility using four types of analytical constraints, including three types of penetration (visualized in Fig. 4) and one contact constraint:

a) Inter-object non-penetration: To prevent two objects i and j from intersecting, we require all transformed surface points of object j to lie outside of object i, as measured by the corresponding SDF:

$$\Phi_i\big(\boldsymbol{T}_i \cdot \boldsymbol{T}_j^{-1} \cdot \boldsymbol{b}\big) \ge 0, \;\forall\, i \neq j, \;\forall\, \boldsymbol{b} \in \mathcal{B}_j, \tag{7}$$

where $\boldsymbol{T}_i \cdot \boldsymbol{T}_j^{-1} \cdot \boldsymbol{b}$ denotes the 3D point $\boldsymbol{b}$ (originally in the frame of object j), transformed to the frame of object i using the similarity transforms $\boldsymbol{T}_i \cdot \boldsymbol{T}_j^{-1}$.

b) Object-environment non-penetration: To prevent objects from penetrating the static environment (*e.g.,* table or ground), we require every object point to remain outside the environment SDF denoted by Φ_{env}:

$$\Phi_{\text{env}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big) \ge 0, \;\forall\, i, \;\forall\, \boldsymbol{b} \in \mathcal{B}_i. \tag{8}$$

c) Object-free-space non-penetration: To ensure object pose hypotheses are consistent with the observed free space (Fig. 4), we impose that object points should not enter regions known to be empty, represented by Φ_{free}:

$$\Phi_{\text{free}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big) \ge 0, \;\forall\, i, \;\forall\, \boldsymbol{b} \in \mathcal{B}_i. \tag{9}$$

d) Contact (no floating objects): Finally, we discourage floating objects by enforcing that each object must be in contact with at least one other object (or the environment) up to a small tolerance δ:

$$\forall\, i,\ \exists\, j \in [N]\backslash\{i\},\ \exists\, \boldsymbol{b} \in \mathcal{B}_i \text{ s.t. } \left|\tilde{\Phi}_j(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b})\right| \leq \delta, \quad (10)$$

where we define the world-frame SDF evaluator

$$\tilde{\Phi}_j(\boldsymbol{y}) := \begin{cases} \Phi_{\text{env}}(\boldsymbol{y}), & j = 0, \\ \Phi_j(\boldsymbol{T}_j \cdot \boldsymbol{y}), & j \in [N]. \end{cases} \quad (11)$$

C. Physics-Constrained Pose Estimator

We now rewrite the maximum likelihood pose estimator (1) using the expression we derived for the objective in eqs. (5)-(6) and the constraints in eqs. (7-10). This gives the following physics-constrained pose estimation problem:

$$\begin{aligned} \underset{\{\boldsymbol{T}_i\}_{i=1}^N \in \text{SIM}(3)^N}{\arg\min} \quad & \sum_{i=1}^{N} \sum_{j=1}^{M_{A,i}} d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \mathcal{B}_i) \\ \text{s.t.} \quad & \Phi_i\big(\boldsymbol{T}_i \cdot \boldsymbol{T}_j^{-1} \cdot \boldsymbol{b}\big) \geq 0, \forall\, i \neq j,\ \forall\, \boldsymbol{b} \in \mathcal{B}_j, \\ & \Phi_{\text{env}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big) \geq 0, \forall\, i,\ \forall\, \boldsymbol{b} \in \mathcal{B}_i \\ & \Phi_{\text{free}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big) \geq 0, \forall\, i,\ \forall\, \boldsymbol{b} \in \mathcal{B}_i, \\ & \min_{j \in [N]\backslash\{i\}} \min_{\boldsymbol{b} \in \mathcal{B}_i} \left|\tilde{\Phi}_j(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b})\right| \leq \delta, \forall\, i. \end{aligned} \quad (12)$$

Problem (12) is highly nonconvex and optimizes N similarity transformations—one per object. We will solve (12) using rejection sampling: that is, we sample a configuration of objects and enforce physical plausibility by simply rejecting any proposal that violates the constraints in (12). We check inter-object, object-environment, and contact constraints using the library [28]. For object-free-space constraint, we render the object as a depth map and directly compare it with observed depth map. Then, among the samples that satisfy the constraints, we select the one achieving the smallest objective.

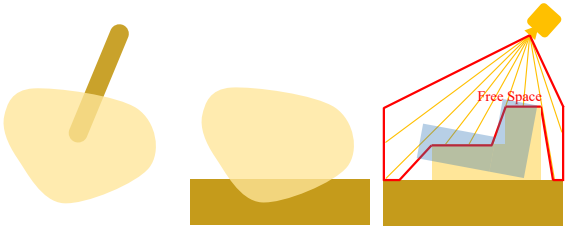
A key remaining challenge is to reduce the dimensionality of the sampling space. Contrary to the single-object case discussed in Remark 1, naively solving Problem (12) via sampling would require high-dimensional samples, with a dimension that grows with the number N of objects in the scene. To address this, we leverage the notion of *contact scene graph* to sequentially estimate each object's similarity transform via sampling.

D. Regaining Tractability via Contact Scene Graph

We start by recalling the classical Contact Scene Graph.

Definition 1 (Contact Scene Graph (CSG) [22]) A *contact scene graph* (CSG) is a tuple $\mathcal{SG} = (\mathcal{G}, \boldsymbol{\theta})$ where $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is an undirected graph and $\boldsymbol{\theta}$ are parameters associated with the nodes of the graph.

In a CSG, the vertices $\mathcal{V} := \{v_0, v_1, \ldots, v_N\}$ represent $N+1$ rigid bodies (v_0 typically represents the environment, *e.g.,* a table for table-top scenes). Each vertex v is associated with a parameter $\boldsymbol{\theta}_v$ which consists of object pose and shape information. An edge $(u, v) \in \mathcal{E}$ indicates that rigid body u is in physical contact with rigid body v. That is, there exist points $\boldsymbol{p}_u, \boldsymbol{p}_v \in \mathbb{R}^3$ on the surface of rigid bodies u and v respectively such that $\boldsymbol{p}_u = \boldsymbol{p}_v$. The CSG is undirected because if u is in contact with v, then v is in contact with u; however, it is not necessarily acyclic (*e.g.,* the set of Jenga blocks in the middle of Fig. 1 form a ring of n nodes).



(a) (b) (c)

Fig. 4: (a) Inter-object penetration. (b) Object-environment penetration. (c) Object-free-space penetration: The estimated object (blue) penetrates the free space (red). Free space from an RGB-D image is defined as the empty volume along each camera ray up to the observed depth surface.

Our goal, as described in the previous sections, is to estimate $\boldsymbol{\theta}_v := (\boldsymbol{T}_v, \boldsymbol{S}_v)$. One could think about a CSG as a graphical model encoding relations between the variables of interest. However, inference on an undirected cyclic graph is computationally hard [11]; on the other hand, inference over a directed acyclic graph (DAG) can be solved efficiently. We leverage this insight below: we first infer the CSG from an RGB image and then approximate it as a DAG; finally, we perform fast inference over the DAG using sampling.

Inferring and Approximating the Topology of the CSG. The topology of the CSG is dictated by which objects are in contact. While a naive approach to infer contact would be to look at the 3D point clouds associated to each object and check if their distance is smaller than a threshold, we found this approach to be quite sensitive to noise and occlusions. For example, two separated objects can be mistakenly inferred as touching due to depth noise/outliers near edges, while true contacts (*e.g.,* object resting on a surface) can be missed entirely because the contact patch is occluded and thus absent from the point cloud. On the other hand, we find that modern Vision-Language Models (VLMs) empirically give accurate answers to queries about contact. Therefore, we use a VLM to infer the topology of the CSG (see Appendix A for details).

After inferring the full CSG, we approximate it with a DAG. We construct the DAG using breadth-first search and choose the environment node as the root. As we traverse the graph, we orient each edge in the order of traversal, yielding a directed graph. This structure enables a *greedy* rollout: we estimate one object at a time (starting from the root) and treat the parents as fixed when inferring descendants. The number of inference calls on the DAG is exactly N rather than $O(N)$ for inference on a full CSG. We emphasize that this is an approximation:

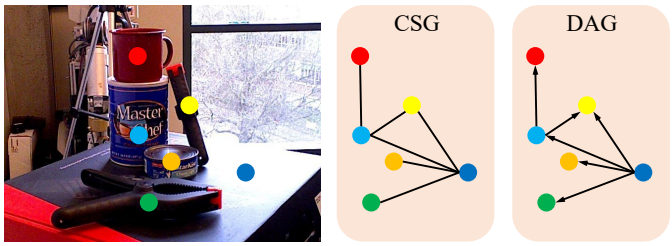


Fig. 5: Sample image and corresponding contact scene graph (CSG) with a directed acyclic graph (DAG) approximation.

greedy rollout is not guaranteed to recover the *global* MLE assignment in (12); however, we observe it achieves excellent accuracy in practice.

Specifically, let $\mathcal{P}(i) \subseteq \mathcal{V}$ be the set of parents of node i. After approximating the graph with a DAG, problem (12) decouples into N subproblems, such that each subproblem is independent given the estimates of their parent nodes:

$$
\begin{aligned}
&\min_{\boldsymbol{T}_i \in \mathrm{SIM}(3)} \quad \sum_{j=1}^{M_{A,i}} d(s_i \boldsymbol{R}_i \boldsymbol{a}_{i,j} + \boldsymbol{t}_i, \mathcal{B}_i) \\
\text{s.t.} \quad &\Phi_i\big(\boldsymbol{T}_i \cdot \boldsymbol{T}_j^{-1} \cdot \boldsymbol{b}\big) \geq 0, \forall\, i \neq j,\ \forall \boldsymbol{b} \in \mathcal{B}_j, j \in \mathcal{P}(i), \\
&\Phi_{\text{env}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big) \geq 0, \forall\, i,\ \forall \boldsymbol{b} \in \mathcal{B}_i, \qquad (13) \\
&\Phi_{\text{free}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big) \geq 0, \forall\, i,\ \forall \boldsymbol{b} \in \mathcal{B}_i, \\
&\min_{j \in \mathcal{P}(i) \setminus \{i\}} \ \min_{\boldsymbol{b} \in \mathcal{B}_i} \big|\tilde{\Phi}_j(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b})\big| \leq \delta, \forall\, i.
\end{aligned}
$$

Therefore, we start by solving for the root of the tree in isolation. Then, following (13), we solve for the immediate children of the root node in isolation, and so on. This is closely related to shock propagation in graphics simulation [21]. Each subproblem in (13) resembles the single-object registration problem in Remark 1, only with additional constraints. This makes it a natural fit for sampling-based optimization. We solve each subproblem via a coarse-to-fine sampling scheme over $\mathrm{SIM}(3)$, as discussed in the next section.

E. Implementation of Coarse-to-Fine Rejection Sampling

We focus on the case where we already have a pose and shape estimator. Our approach takes as input the pose and shape estimate from an existing network (*e.g.,* we use SAM3D [9] and CRISP [46] in our tests) and our sampling scheme corrects that estimate to make it more accurate and physically plausible.

Initialization. We initialize the object translation as the mean of the depth point clouds (restricted to the object mask), after removing outliers via 5% thresholding. We initialize the object scale by comparing the mean depth of the masked point cloud to the mean depth of the rendered point cloud from the network's (*e.g.,* SAM3D) estimate.

Sampler. We solve each decoupled subproblem (13) via a hierarchical sampler over $\mathrm{SIM}(3)$ by separating scale from $\mathrm{SE}(3)$ pose. We first perform a coarse-to-fine search over scale: starting from a broad set of scale hypotheses, we progressively refine around the best-performing candidates. For each scale hypothesis, we then run an $\mathrm{SE}(3)$ registration subroutine that samples translation and rotation in a coarse-to-fine manner. This coarse-to-fine strategy is also used in [20] and resembles BnB. Specifically, at each refinement level, we retain the top candidates according to the loss at the previous level and resample locally around them to increase resolution. We refer the reader to Appendix B for further details.

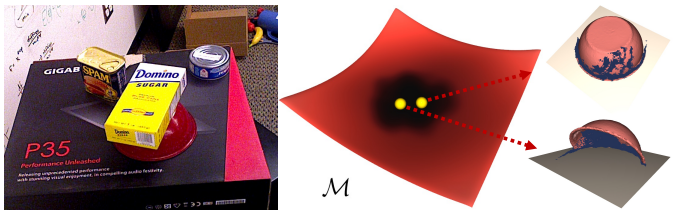


Fig. 6: Conceptual illustration of the loss landscape on the $\mathcal{M} = \mathrm{SE}(3)$ pose manifold for the red bowl shown in the left image. The landscape exhibits a region of similar loss values with two global minima (yellow balls) arising from the ambiguity of partial point cloud observations (dark blue). However, only one of these minima satisfies physics constraints and represents the correct pose. Physics constraints prune the feasible set to identify the physically valid solution.

Physics Constraints. We use libigl [28] for object-object penetration and contact checking. For object-free-space penetration, we use the voxel-based depth renderer implemented in [39]. We apply rejection sampling to the top $B = 16$ candidates ranked by Chamfer loss. If all B violate constraints, we return the best-scoring pose. A visualization of how physics constraints can help pose estimation is shown in Fig. 6.

Robust Loss. In practice, we augment the objective in (13) with a robust loss to mitigate the effect of segmentation artifacts, shape prediction errors, and depth sensor noise. In particular, we use the Geman-McClure robust loss [3] to robustify the Chamfer distance, namely $\rho_{\text{GM}}\big(d(s_i\, \boldsymbol{R}_i\, \boldsymbol{a}_{i,j} + \boldsymbol{t}_i,\ \mathcal{B}_i)\big)$, where $\rho_{\text{GM}}(d) = d^2/(d^2 + \delta^2), \delta = 0.05\,\text{m}$.

System. Overall, this results in a unified, sampling-based solver for physics-constrained $\mathrm{SIM}(3)$ registration, which we name *Picasso*. Picasso is a pose corrector, as it leverages rough poses for object scale initialization. Picasso enables us to estimate each object in a fraction of a second using GPU-optimized parallel sampling (Section V-A), and outperforms state of the art estimators (Section V).

Takeaway

Picasso (i) decouples holistic scene understanding into a sequential estimation of object poses, while accounting for physical constraints among objects, and (ii) enables fast inference using GPU-parallelized coarse-to-fine rejection sampling.

IV. Picasso Dataset and Physics Metric

We now introduce the *Picasso dataset*, a new object-centric dataset focused on contact-rich real-world scenes. We also define a *scene plausibility score* (SPS), which is designed to measure the physical plausibility of a scene reconstruction.

A. The Picasso Dataset

Motivation. Object-centric datasets provide ground-truth annotations for tasks such as object segmentation, pose, and shape. Large-scale object shape repositories are now widely available [8, 12, 43, 54, 14], and several datasets offer pose and shape annotations, including NOCS and YCB-Video [49, 56]. However, most existing datasets do not model physical interactions and contacts explicitly. The closest related effort is Vysics [5], which focuses on contact-rich tabletop trajectories. Our Picasso dataset provides not only object shapes but also per-object 6D pose annotations, and targets RGB-D single-image estimation rather than object tracking.

Dataset Annotation. We capture RGB-D data using an iPad's built-in LiDAR and camera sensors at 1920×1440 resolution and 60 Hz. Object masks are produced automatically using SAM2 [42] and then verified and corrected by human annotators. Subsequently, the object models are scanned and refined in Blender [10]. To obtain 3D poses, we solve PnP by labeling 2D keypoints in the images and establishing correspondences to keypoints on the scanned 3D object models. Finally, we refine the raw depth maps by rendering the object models under the annotated poses.

Dataset Characteristics. The Picasso dataset contains 10 contact-rich static scenes featuring 10 everyday objects (Fig. 1). For each scene, we select 10 images and annotate the pose and shape of every object. The number of objects per scene ranges from two to eight. The scenes cover diverse contact types, including point contacts (*e.g.,* mugs supported by a rack), line contacts (*e.g.,* a spatula resting on a pan), and flush contacts (*e.g.,* stacked Jenga blocks).

B. Physics Metric

We define the *Scene Plausibility Score (SPS)* to ascertain scene-level physics plausibility. Given a set of pose and shape estimates of the scene, we create a digital twin in simulation[2] [53] and run a short rollout of T steps. For all scenes, we use $T = 20$ with a simulation timestep of $1/240s$. Then we compute SPS:

$$S^{SPS} := \frac{1}{2} m \boldsymbol{v}^\top \boldsymbol{v} + \frac{1}{2} \boldsymbol{\omega}^\top \boldsymbol{J} \boldsymbol{\omega} \tag{14}$$

where $\boldsymbol{v}$ and $\boldsymbol{\omega}$ are the linear and angular velocity of the object. We set nominal mass $m = 1\,\text{kg}$ and inertia $\boldsymbol{J} = \mathbf{I}_3$ (identity matrix). The metric measures the kinetic energy of a system with this canonical mass and inertia. Due to potential inaccuracies in the simulator's physical modeling, we threshold translational and rotational kinetic energies to a maximum of 10 $kg \cdot m^2/s^2$ in all experiments.

[2]We used PyBullet in this paper but any other simulator can be used.

TABLE I: Evaluation results on the ADD-S (m) and ADD-S (AUC) metrics for the YCB-V dataset. Best, Second-best.

	ADD-S ↓		ADD-S (AUC) ↑		
Method	Mean	Median	1 cm	2 cm	3 cm
CosyPose [31]	0.010	0.007	0.30	0.56	0.68
BundleSDF [51]	0.014	0.012	0.14	0.37	0.55
GDRNet++ [34]	0.013	0.011	0.22	0.43	0.58
CRISP-Real [46]	0.009	0.004	0.44	0.58	0.75
CRISP-Syn [46]	0.015	0.010	0.21	0.41	0.55
CRISP-Syn+GD	0.015	0.011	0.21	0.41	0.55
CRISP-Syn+GICP	0.012	0.003	0.44	0.58	0.67
CRISP-Syn+Picasso	0.008	0.003	0.52	0.69	0.77
CRISP-Real+Picasso	0.008	0.003	0.53	0.70	0.78

TABLE II: Evaluation results on the SPS, NPS (mm), observable correctness, and runtime on the YCB-V dataset. Best, Second-best.

Method	SPS ↓	NPS ↓	OC ↑	run time (s) ↓
CRISP-Real	9.05	4.57	36%	/
CRISP-Syn	9.36	6.38	15%	/
CRISP-Syn+GICP	9.56	4.70	42%	0.03
CRISP-Syn+Picasso	8.71	3.99	55%	0.35
CRISP-Real+Picasso	8.77	3.95	55%	0.33

V. Experiments

We evaluate Picasso in a variety of real-world settings and retrofit it to CRISP [46] and SAM3D [9]. Section V-A shows that Picasso bridges the sim-to-real gap for CRISP, allowing a simulation-trained approach to outperform an approach trained on real data. Section V-B shows Picasso improves SAM3D performance on the YCB-V and Picasso datasets.

A. Boosting CRISP's Performance Using Picasso

Setup. We train two models: CRISP-Real, which is trained on the real-world training images provided by the YCB-V dataset, and CRISP-Syn, which is trained on 4,200 synthetic images (200 per object) rendered with BlenderProc [13] as provided by [46]. We use the synthetic setup to assess Picasso's ability to correct rough estimates caused by a large sim-to-real gap. In addition to the two CRISP variants, we compare against two baselines: (i) A physics-guided approach that refines CRISP by incorporating non-penetration losses, optimized via gradient descent (more details in Appendix H), (ii) a Generalized ICP-based pose corrector, as in [30]. We report ADD-S [56], observable correctness (OC) [46], and the proposed Scene Plausibility Score (SPS). We also report the Non-Penetration Score (NPS) [62] with a small modification to exclude bad shape estimates (Appendix C).

Results. As shown in Table I, CRISP-Real outperforms [31, 51, 34] and CRISP-Syn, indicating a large sim-to-real gap. With Picasso, however, CRISP-Syn becomes the top-performing method, outperforming local pose refinement such as gradient descent and GICP [44], and even surpassing CRISP-Real to close the sim-to-real gap. Table II shows that Picasso also delivers the best results on physics metrics (NPS and SPS) and achieves fast performance with runtime under 1 second.

TABLE III: Comparison of methods across 12 YCB-Video trajectories. ADD-S (mm), SPS (physics loss), NPS (mm). S: SAM3D. S+P-w/o PC: SAM3D+Picasso w/o physics constraints. S+P: SAM3D+Picasso. Best.

Trajectory	ADD-S ↓			SPS ↓			NPS ↓		
	S[9]	S+P -w/o PC	S+P	S[9]	S+P -w/o PC	S+P	S[9]	S+P -w/o PC	S+P
traj_48	16.97	8.98	9.06	15.32	5.38	5.16	10.13	2.82	2.76
traj_49	29.53	26.74	26.59	4.21	2.32	2.14	7.94	5.03	4.72
traj_50	11.77	19.91	20.48	1.60	0.86	1.14	5.65	3.86	3.53
traj_51	13.25	7.40	7.84	8.51	2.87	3.01	7.31	2.37	2.13
traj_52	17.78	6.28	6.76	6.16	0.12	0.12	16.53	3.20	2.91
traj_53	16.07	5.17	5.46	6.50	0.04	0.04	11.51	3.23	2.98
traj_54	22.94	10.22	10.19	10.77	4.53	4.69	10.70	2.51	2.47
traj_55	68.73	9.53	10.23	3.60	0.63	0.63	11.11	2.16	1.95
traj_56	16.74	9.21	9.45	4.42	1.48	1.19	9.01	2.77	2.67
traj_57	19.48	6.88	6.95	2.08	0.48	0.39	5.12	1.94	1.69
traj_58	13.06	7.11	7.31	3.43	3.18	2.73	10.27	4.16	3.86
traj_59	13.81	6.11	6.46	3.54	1.03	1.02	6.26	1.96	1.84
Overall	21.68	10.29	10.56	5.84	1.91	1.85	9.29	3.00	2.79

B. Boosting SAM3D's Performance Using Picasso

We compare against SAM3D [9], a foundation model that predicts object shape and 6D pose (up to scale) from monocular images. To estimate scale for SAM3D, we compute the ratio between the mean depth of the masked depth map and the mean rendered depth from SAM3D's predicted pose and shape. For Picasso, we take the SAM3D estimates and compute a similarity transform as discussed in Section III.

1) YCB-V Dataset: We test on the YCB-V dataset from the BOP19 challenge consisting of 900 frames.

Results. We compare SAM3D (S) and SAM3D+Picasso (S+P) in Table III. We also compare with SAM3D+Picasso without physics constraints (S+P-w/o PC) (*i.e.,* only sampling from $\mathrm{SIM}(3)$ to minimize the objective). Picasso significantly improves SAM3D in both a geometric metric (ADD-S) and physical metric (NPS and SPS). Picasso also outperforms the ablation without physics constraints, but the benefit is more modest because we only check physics plausibility on a small buffer of candidates ranked by Chamfer loss.

2) Picasso Dataset: We further test on the Picasso dataset, which contains 100 frames. Specifically, we compare against: (1) SAM3D with RGB input (denoted as SAM3D), (2) SAM3D with both RGB and pointmap input (denoted as SAM3D-PM), and (3) SAM3D-PM combined with GICP.

Results. Fig. 7 shows qualitative 3D reconstructions. While SAM3D produces severe inter-object penetrations and counter-intuitive floating objects, Picasso yields physically valid results. As shown in Table IV, Picasso consistently improves performance across all metrics under both the SAM3D and SAM3D-PM settings by a large margin, and it outperforms the registration method GICP on all but one metric.

C. User Study: Alignment with Physics Metrics

Finally, we investigate the question: *Do Picasso's reconstructions align with human physical intuition?* Towards this goal, we perform a user study using Amazon Mechanical Turk, where users rank the plausibility of 3D reconstructions given videos of the corresponding digital twins. The results show that (i) human evaluation aligns well with our Scene Plausibility Score (SPS), and (ii) Picasso's reconstructions are ranked consistently better than SAM3D results in terms of physical plausibility. We refer the reader to Appendix J for details about the data collection and results.

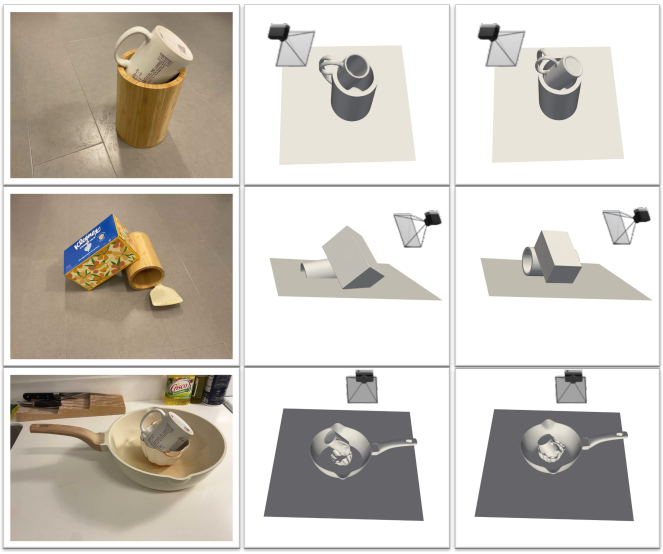


Fig. 7: Visualization of 3D scene reconstruction from the Picasso Dataset. **Left:** Input RGB images. **Middle:** SAM3D reconstructions. **Right:** SAM3D+Picasso reconstructions. While SAM3D reconstructions have severe penetrations and floating objects, Picasso produces physically plausible configurations.

TABLE IV: Comparison of methods across 10 Picasso sequences. ADD-S (mm), SPS (physics loss), NPS (mm). SAM3D: SAM3D with RGB input, SAM3D-PM: SAM3D with both RGB and pointmap input. Best, Second-best.

Method	ADD-S (mm) ↓		ADD-S (AUC) ↑			SPS ↓	NPS ↓
	Mean	Median	1 cm	2 cm	3 cm		
SAM3D-PM	11.45	6.49	33.59	56.17	68.35	8.91	5.67
+GICP	7.23	5.77	43.82	69.04	78.93	8.58	4.32
+Picasso (Ours)	7.84	5.18	47.07	70.99	79.73	6.59	3.50
SAM3D	11.71	7.52	30.60	53.61	66.53	8.50	5.23
+Picasso (Ours)	7.57	4.95	47.38	70.91	79.77	7.63	3.74

VI. Limitations and Future Work

We conclude by outlining the limitations of Picasso and highlighting several promising directions for future research:

a) Alternative Objectives: Although we focus on point-cloud registration, physics-constrained sampling naturally extends to other highly nonconvex objectives. For instance, it can be combined with semantic-based or rendering-based losses [32, 20, 64, 52], or with latent-space objectives that compare observations and renderings through learned embeddings [15, 4].

b) Accelerating the Solver: We plan to investigate faster sampling-based solvers, for example by leveraging BnB and 3D Euclidean Distance Transforms for efficient scoring [58].

c) Improving Contact Scene Graph Inference: Our current approach makes the contact graph acyclic and performs local MLE rollout. Future work will relax this approximation to support mutual interactions between upstream and downstream nodes, while preserving fast inference [21].

d) Physics-Constrained Foundation Models: We applied Picasso to improve inference-time performance of foundation models, such as SAM3D. Beyond inference, another direction is to use Picasso to generate constraint-satisfying data that could be used to augment training for foundation models. In addition, such physics constraints could be incorporated directly into the model training procedure, building on recent work in constrained diffusion modeling [17, 37, 6].

VII. Conclusion

We proposed an approach for holistic scene understanding, which accounts for sensor measurements (*e.g.,* RGB-D scans) and physical plausibility. The proposed approach, Picasso, decouples scene reconstruction into a sequential physics-informed estimation of object poses, and enables fast inference using GPU-parallelized coarse-to-fine rejection sampling. We also proposed the *Picasso dataset*, a collection of 10 contact-rich real-world scenes with ground truth annotations, as well as a novel metric to quantify physical plausibility, which we open-sourced with our benchmark. Results on the Picasso and YCB-V dataset show that Picasso largely outperforms the state of the art while providing reconstructions that are both physically plausible and more aligned with human intuition.

References

[1] Aditya Agarwal, Gaurav Singh, Bipasha Sen, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Scenecomplete: Open-world 3d scene completion in cluttered real world environments for robot manipulation. *IEEE Robotics and Automation Letters*, 11(1):482–489, 2025.

[2] William Agnew, Christopher Xie, Aaron Walsman, Octavian Murad, Yubo Wang, Pedro Domingos, and Siddhartha Srinivasa. Amodal 3d reconstruction for robotic manipulation via stability and connectivity. In *Conference on Robot Learning (CoRL)*, pages 1498–1508. PMLR, 2021.

[3] Jonathan T Barron. A general and adaptive robust loss function. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4331–4339, 2019.

[4] Matthias Bauer, Emilien Dupont, Andy Brock, Dan Rosenbaum, Jonathan Richard Schwarz, and Hyunjik Kim. Spatial functa: Scaling functa to imagenet classification and generation. *arXiv preprint arXiv:2302.03130*, 2023.

[5] Bibit Bianchini, Minghan Zhu, Mengti Sun, Bowen Jiang, Camillo J Taylor, and Michael Posa. Vysics: Object reconstruction under occlusion by fusing vision and contact-rich physics. *arXiv preprint arXiv:2504.18719*, 2025.

[6] Matthieu Blanke, Yongquan Qu, Sara Shamekh, and Pierre Gentine. Strictly constrained generative modeling via split augmented langevin sampling. *arXiv preprint arXiv:2505.18017*, 2025.

[7] T. M. Breuel. Implementation techniques for geometric branch-and-bound matching methods. *Comput. Vis. Image Underst.*, 90(3):258–294, 2003.

[8] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[9] 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. *arXiv preprint arXiv:2511.16624*, 2025.

[10] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL http://www.blender.org.

[11] G.F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42(2-3):393–405, 1990. ISSN 0004-3702.

[12] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems (NeurIPS)*, 36: 35799–35813, 2023.

[13] Maximilian Denninger, Martin Sundermeyer, Dominik Winkelbauer, Dmitry Olefir, Tomas Hodan, Youssef Zidan, Mohamad Elbadrawy, Markus Knauer, Harinandan Katam, and Ahsan Lodhi. Blenderproc: Reducing the reality gap with photorealistic rendering. In *16th Robotics: Science and Systems, RSS 2020, Workshops*, 2020.

[14] 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 *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 2553–2560. IEEE, 2022.

[15] Emilien Dupont, Hyunjik Kim, SM Eslami, Danilo Rezende, and Dan Rosenbaum. From data to functa: Your data point is a function and you can treat it like one. *arXiv preprint arXiv:2201.12204*, 2022.

[16] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. ACM*, 24: 381–395, 1981.

[17] Nic Fishman, Leo Klarner, Valentin De Bortoli, Emile Mathieu, and Michael Hutchinson. Diffusion models for constrained domains. *arXiv preprint arXiv:2304.05364*, 2023.

[18] Yang Fu and Xiaolong Wang. Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and a new dataset. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:27469–27483, 2022.

[19] N. Gothoskar, M. Cusumano-Towner, B. Zinberg,

M. Ghavamizadeh, F. Pollok, A. Garrett, J.B. Tenenbaum, D. Gutfreund, and V.K. Mansinghka. 3DP3: 3D scene perception via probabilistic programming. In *arXiv preprint: 2111.00312*, 2021.

[20] Nishad Gothoskar, Matin Ghavami, Eric Li, Aidan Curtis, Michael Noseworthy, Karen Chung, Brian Patton, William T Freeman, Joshua B Tenenbaum, Mirko Klukas, et al. Bayes3d: fast learning and inference in structured generative models of 3d objects and scenes. *arXiv preprint arXiv:2312.08715*, 2023.

[21] Eran Guendelman, Robert Bridson, and Ronald Fedkiw. Nonconvex rigid bodies with stacking. *ACM transactions on graphics (TOG)*, 22(3):871–878, 2003.

[22] James K Hahn. Realistic animation of rigid bodies. *ACM Siggraph computer graphics*, 22(4):299–308, 1988.

[23] R.I. Hartley and F. Kahl. Global optimization through rotation space search. *Intl. J. of Computer Vision*, 82(1): 64–79, 2009.

[24] Binbin Huang, Haobin Duan, Yiqun Zhao, Zibo Zhao, Yi Ma, and Shenghua Gao. Cupid: Generative 3d reconstruction via joint object and pose modeling. *arXiv preprint arXiv:2510.20776*, 2025.

[25] Shun Iwase, Katherine Liu, Vitor Guizilini, Adrien Gaidon, Kris Kitani, Rareş Ambruş, and Sergey Zakharov. Zero-shot multi-object scene completion. In *European Conf. on Computer Vision (ECCV)*, pages 96–113. Springer, 2024.

[26] Shun Iwase, Muhammad Zubair Irshad, Katherine Liu, Vitor Guizilini, Robert Lee, Takuya Ikeda, Ayako Amma, Koichi Nishiwaki, Kris Kitani, Rares Ambrus, et al. Zerograsp: Zero-shot shape reconstruction enabled robotic grasping. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 17405–17415, 2025.

[27] G. Izatt, H. Dai, and R. Tedrake. Globally optimal object pose estimation in point clouds with mixed-integer programming. In *Proc. of the Intl. Symp. of Robotics Research (ISRR)*, 2017.

[28] Alec Jacobson, Daniele Panozzo, et al. libigl: A simple C++ geometry processing library, 2018. https://libigl.github.io/.

[29] Hanxiao Jiang, Hao-Yu Hsu, Kaifeng Zhang, Hsin-Ni Yu, Shenlong Wang, and Yunzhu Li. Phystwin: Physics-informed reconstruction and simulation of deformable objects from videos. *arXiv preprint arXiv:2503.17973*, 2025.

[30] Kenji Koide. small_gicp: Efficient and parallel algorithms for point cloud registration. *Journal of Open Source Software*, 9(100):6948, August 2024. doi: 10.21105/joss.06948.

[31] Y. Labbe, J. Carpentier, M. Aubry, and J. Sivic. CosyPose: Consistent multi-view multi-object 6D pose estimation. In *European Conf. on Computer Vision (ECCV)*, 2020.

[32] 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. 2022.

[33] H. Lim, D. Kim, G. Shin, J. Shi, I. Vizzo, H. Myung, J. Park, and L. Carlone. KISS-Matcher: Fast and robust point cloud registration revisited. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2025.

[34] Xingyu Liu, Ruida Zhang, Chenyangguang Zhang, Bowen Fu, Jiwen Tang, Xiquan Liang, Jingyi Tang, Xiaotian Cheng, Yukang Zhang, Gu Wang, and Xiangyang Ji. Gdrnpp. https://github.com/shanice-l/gdrnpp_bop2022, 2022.

[35] Martin Malenickỳ, Martin Cífka, Médéric Fourmy, Louis Montaut, Justin Carpentier, Josef Sivic, and Vladimir Petrik. Physpose: Refining 6d object poses with physical constraints. *arXiv preprint arXiv:2503.23587*, 2025.

[36] 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. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:25747–25780, 2024.

[37] Adam Nordenhög and Akash Sharma. Score-based constrained generative modeling via langevin diffusions with boundary conditions. *arXiv preprint arXiv:2510.23985*, 2025.

[38] Á. Parra Bustos, T. J. Chin, and D. Suter. Fast rotation search with stereographic projections for 3d registration. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3930–3937, 2014.

[39] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[40] G. Pavlakos, X. Zhou, A. Chan, K. Derpanis, and K. Daniilidis. 6-dof object pose from semantic keypoints. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[41] Li Puyin, Tiange Xiang, Ella Mao, Shirley Wei, Xinye Chen, Adnan Masood, Li Fei-Fei, and Ehsan Adeli. Quantiphy: A quantitative benchmark evaluating physical reasoning abilities of vision-language models. *arXiv preprint arXiv:2512.19526*, 2025.

[42] N. Ravi, V. Gabeur, Y-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K.V. Alwala, N. Carion, C-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. SAM 2: Segment anything in images and videos, 2024. URL https://arxiv.org/abs/2408.00714.

[43] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Intl. Conf. on Computer Vision (ICCV)*, pages 10901–10911, 2021.

[44] Aleksandr Segal, Dirk Haehnel, and Sebastian Thrun.

Generalized ICP. In *Robotics: Science and Systems (RSS)*, Jun. 2009. doi: 10.15607/RSS.2009.V.021.

[45] J. Shi*, R. Talak*, D. Maggio, and L. Carlone. A correct-and-certify approach to self-supervise object pose estimators via ensemble self-training. In *Robotics: Science and Systems (RSS)*, 2023. https://arxiv.org/pdf/2302.06019.pdf.

[46] J. Shi, R. Talak, H. Zhang, D. Jin, and L. Carlone. CRISP: Object pose and shape estimation with test-time adaptation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025. .

[47] R. Talak, L. Peng, and L. Carlone. Certifiable 3D object pose estimation: Foundations, learning models, and self-training. *IEEE Trans. Robotics*, 39(4):2805–2824, 2023. https://arxiv.org/pdf/2206.11215.pdf.

[48] Meng Tian, Marcelo H Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *European Conf. on Computer Vision (ECCV)*, pages 530–546. Springer, 2020.

[49] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2651, 2019.

[50] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 5294–5306, 2025.

[51] Bowen Wen, Jonathan Tremblay, Valts Blukis, Stephen Tyree, Thomas Müller, Alex Evans, Dieter Fox, Jan Kautz, and Stan Birchfield. Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 606–617, 2023.

[52] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. FoundationPose: Unified 6D Pose Estimation and Tracking of Novel Objects . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17868–17879, Los Alamitos, CA, USA, June 2024. IEEE Computer Society. doi: 10.1109/CVPR52733.2024.01692. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.01692.

[53] Jiajun Wu, Ilker Yildirim, Joseph J Lim, Bill Freeman, and Josh Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 28, 2015.

[54] Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 803–814, 2023.

[55] Hongchi Xia, Chih-Hao Lin, Hao-Yu Hsu, Quentin Leboutet, Katelyn Gao, Michael Paulitsch, Benjamin Ummenhofer, and Shenlong Wang. Holoscene: Simulation-ready interactive 3d worlds from a single video. *arXiv preprint arXiv:2510.05560*, 2025.

[56] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018.

[57] H. Yang, J. Shi, and L. Carlone. TEASER: Fast and Certifiable Point Cloud Registration. *IEEE Trans. Robotics*, 37(2):314–333, 2020. extended arXiv version 2001.07715 https://arxiv.org/pdf/2001.07715.pdf.

[58] J. Yang, H. Li, D. Campbell, and Y. Jia. Go-ICP: A globally optimal solution to 3D ICP point-set registration. *IEEE Trans. Pattern Anal. Machine Intell.*, 38(11):2241–2254, November 2016. ISSN 0162-8828.

[59] Wen Yang, Zhixian Xie, Xuechao Zhang, Heni Ben Amor, Shan Lin, and Wanxin Jin. Twintrack: Bridging vision and contact physics for real-time tracking of unknown dynamic objects. *arXiv preprint arXiv:2505.22882*, 2025.

[60] 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 (TOG)*, 44(4):1–19, 2025.

[61] Xihang Yu, Rajat Talak, Jingnan Shi, Ulrich Viereck, Igor Gilitschenski, and Luca Carlone. Box pose and shape estimation and domain adaptation for large-scale warehouse automation. *arXiv preprint arXiv:2507.00984*, 2025.

[62] Mengchao Zhang and Kris Hauser. Non-penetration iterative closest points for single-view multi-object 6d pose estimation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1520–1526. IEEE, 2022.

[63] B. Zheng, Y. Zhao, J. C. Yu, K. Ikeuchi, and S. Zhu. Beyond point clouds: Scene understanding by reasoning geometry and physics. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3127–3134, 2013.

[64] Guangyao Zhou, Nishad Gothoskar, Lirui Wang, Joshua B Tenenbaum, Dan Gutfreund, Miguel Lázaro-Gredilla, Dileep George, and Vikash K Mansinghka. 3d neural embedding likelihood: Probabilistic inverse graphics for robust 6d pose estimation. In *Intl. Conf. on Computer Vision (ICCV)*, pages 21625–21636, 2023.

[65] Minghan Zhu, Zhiyi Wang, Qihang Sun, Maani Ghaffari, and Michael Posa. Object reconstruction under occlusion with generative priors and contact-induced constraints. *arXiv preprint arXiv:2512.05079*, 2025.

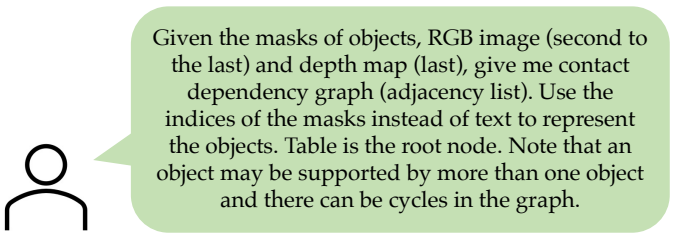


Fig. 8: VLM prompt for contact scene graph generation.

Level	Type	$\theta_{\max}$	θ_{step}	# rotations per seed
1	Global	–	–	1024
2	Tangent Space	30°	6°	515
3	Tangent Space	6°	2°	123
4	Tangent Space	3°	2°	19

TABLE V: Hierarchical rotation sampling schedule. Level 1 performs global sampling on $\mathrm{SO}(3)$; later levels refine locally around propagated seeds.

APPENDIX

A. VLM for CSG Generation

In this section, we explain how we construct a contact scene graph (CSG) for an arbitrary scene using a vision-language model (VLM). Empirically, we find that VLMs can accurately estimate object contacts using only a single RGB image. We provide a VLM with object masks, the RGB image, and the depth map, and we query it for a contact graph between the object masks. We use the prompt given in Fig. 8. We use Google Gemini throughout all experiments.

B. Coarse-to-Fine Sampling Details

As we mentioned in Section III-E, we first perform a coarse-to-fine search to estimate scale, rotation, and translation. For each scale hypothesis, we run an $\mathrm{SE}(3)$ registration subroutine that jointly samples translation and rotation, progressively refining and sampling around the best-performing candidates.

To find scale, we use three levels spanning $[0.5, 1.1]$ with $K = 5$ uniform samples per level, and refine each subsequent level around the best candidate. For rotation, we adopt four levels. At the coarsest level, we first sample 1024 rotations uniformly on $\mathrm{SO}(3)$. At subsequent levels, we take seed rotations from the top candidates of the previous level and refine locally. Given a seed rotation $\boldsymbol{R}_{\text{seed}}$, we sample local perturbations in the tangent space. That is, $\boldsymbol{R}_{\text{local}} = \boldsymbol{R}_{\text{seed}} \cdot \exp(\boldsymbol{\omega})$, where $\exp(\cdot)$ is the exponential map and $\boldsymbol{\omega} \in \mathbb{R}^3$ is a tangent space vector with elements $\omega_i \in [-\theta_{\max}, \theta_{\max}]$. Then we conduct a local grid search with step size θ_{step}. The local search window $(\theta_{\max}, \theta_{\text{step}})$ decreases per level and gets finer as shown in Table V. Lastly, for translation we normalize the object to unit scale using the predicted shape and estimate translation in this normalized frame. At the coarser level (level 1), we evaluate a uniform $20^3 = 8000$ grid over the x, y, and z range $[-0.5, 0.5]$ (in normalized coordinates). For finer levels (levels 2-4), we perform coarse-to-fine refinement around the best unnormalized translation using 5 samples per dimension, with level-dependent spacing $\tau_\ell \in \{0.05, 0.01, 0.005\}$ m. The most computationally intensive step occurs at the coarsest stage, where global $\mathrm{SE}(3)$ jointly samples 1024 rotation hypotheses and 8000 translation hypotheses, resulting in approximately 8×10^6 candidates. We use the same hierarchical search schedule for all the experiments.

C. Non-Penetration Score (NPS)

As mentioned in Section III, three types of penetration are considered: inter-object penetration, object-environment penetration (e.g., table/ground), object-free-space penetration. For each object $i \in [N]$, let $\mathcal{J}(i) = \{0\} \cup \big([N] \setminus \{i\}\big) \cup \{\text{free}\}$ denote the set of all surrounding entities for object i, where 0 represents the environment and free represents free space. Let $p(i, j)$ return the deepest penetration between object $i \in [N]$ and $j \in \mathcal{J}(i)$:

$$p(i,j) \;=\; \begin{cases} \mathrm{ReLU}\Big(-\min\Big(\min\limits_{\boldsymbol{b} \in \mathcal{B}_j} \Phi_i\big(\boldsymbol{T}_i \cdot \boldsymbol{T}_j^{-1} \cdot \boldsymbol{b}\big), \\ \qquad \min\limits_{\boldsymbol{b} \in \mathcal{B}_i} \Phi_j\big(\boldsymbol{T}_j \cdot \boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big)\Big)\Big), & j \in [N], \\ \mathrm{ReLU}\Big(-\min\limits_{\boldsymbol{b} \in \mathcal{B}_i} \Phi_{\text{env}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big)\Big), & j = 0, \\ \mathrm{ReLU}\Big(-\min\limits_{\boldsymbol{b} \in \mathcal{B}_i} \Phi_{\text{free}}\big(\boldsymbol{T}_i^{-1} \cdot \boldsymbol{b}\big)\Big), & j = \text{free}. \end{cases} \tag{15}$$

Following [62], the Non-Penetration Score (NPS) for object i is

$$S_i^{NPS-ND} := \frac{1}{|\mathcal{J}(i)|} \sum_{j \in \mathcal{J}(i)} p(i,j), \tag{16}$$

which evaluates the mean penetration across all objects in the scene. This definition works well as a score for object i if the shape and pose of each object j is accurately estimated. However, a single neighboring object with inaccurate shape or pose can significantly inflate the penetration metric for an accurately estimated object i. To mitigate this issue, we propose an adaptive alternative that excludes neighbors with large shape and pose errors:

$$S_i^{NPS} := \frac{1}{|\mathcal{J}(i)|} \sum_{j \in \mathcal{J}(i)} \mathcal{I}(j) p(i,j), \tag{17}$$

where $\mathcal{I}(j)$ is an indicator function of whether object j is included in the evaluation. For the experiment, we use ADD-S between as the indicator function because it encodes both shape and pose error of an object [56]. We apply a threshold of 0.05 meters across all experiments. Additionally, due to noise in the depth maps which may affect free space estimation, we cap the maximum allowed penetration depth at 0.01 meters throughout all experiments.

D. Additional Details for Picasso Dataset

Fig. 9 shows additional examples from the Picasso dataset. Picasso contains 10 contact-rich static scenes with 10 commonplace objects.

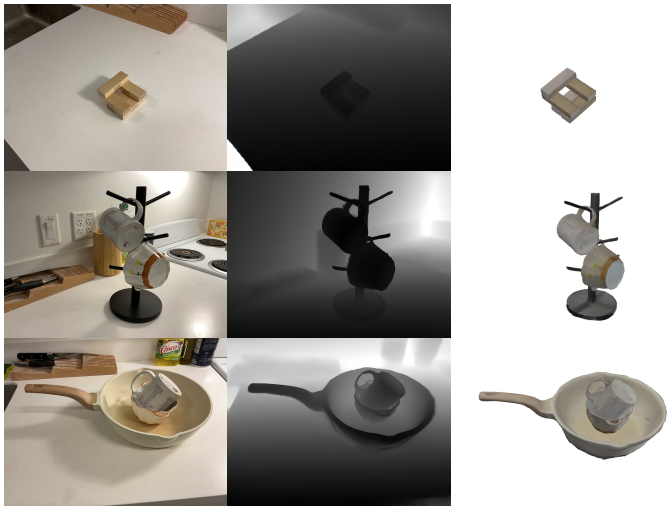


Fig. 9: Examples of Picasso Dataset. **Left:** RGB images. **Middle:** Depth maps. **Right:** Reconstructed 3D models.

TABLE VI: Evaluation results on the ADD-S ($\times 10^{-3}$ m) and ADD-S (AUC %) metrics for the YCB-V dataset. CRISP-Syn+Picasso w/o phy: CRISP-Syn+Picasso with physics constraints turned off. Best.

	ADD-S ↓		ADD-S (AUC %) ↑		
Method	Mean	Median	1 cm	2 cm	3 cm
CRISP-Syn +Picasso w/o phy	8.35	3.04	51.8	68.9	77.1
CRISP-Syn +Picasso	8.28	2.98	51.9	69.1	77.2

E. Ablation Study

Tables VI and VII show the ablation with physics constraints turned off on the YCB-V dataset in the CRISP experiments. The constraint-free version of Picasso achieves faster performance while full Picasso performs better both geometrically and in terms of physical plausibility. We also evaluate constraint satisfaction rates. Picasso with the physics corrector achieves satisfaction rates of 72.34%, whereas the satisfaction rate drops to 52.25% without the physics correction.

Table VIII shows the effect of each constraint term on Picasso. Inter-object and contact constraints play a bigger role than rendering constraints like object-free-space. The effect of individual constraint terms appears to be dataset-dependent, likely because their contributions depend on visibility and ambiguity in the observed point clouds. On YCB-V, the object-environment constraint produces the largest performance gain, while the object-free-space constraint has little measurable effect. In contrast, on IC-BIN, object-free-space constraints are more influential: when the observed point cloud admits two plausible fits, the constraint helps reject the solution that would violate free space as shown in Figure 10.

F. Hardware setup

All experiments were performed on a machine with an NVIDIA RTX 4090 GPU.

TABLE VII: Evaluation results on the SPS, NPS ($\times 10^{-3}$ m), observable correctness (OC), and run time for the YCB-V dataset. CRISP-Syn+Picasso w/o phy: CRISP-Syn+Picasso with physics constraints turned off. Best.

Method	SPS ↓	NPS ↓	OC ↑	run time (s) ↓
CRISP-Syn +Picasso w/o phy	8.76	4.06	54%	0.29
CRISP-Syn +Picasso	8.71	3.99	55%	0.35

TABLE VIII: Ablation studies on YCB-V and IC-BIN datasets. Best.

Method	OC ↑
w/o object-environment	54.42%
w/o inter-object	55.76%
w/o object-free-space	56.17%
w/o contact	54.88%
Picasso	56.17%

YCB-V

Method	ADD-S (mm) ↓
w/o object-environment	8.58
w/o inter-object	8.65
w/o object-free-space	8.69
w/o contact	8.59
Picasso	8.58

IC-BIN

G. Boosting CRISP's Shape Prediction

Tables IX and X show the shape correction capability of Picasso with CRISP. CRISP uses latent shape code for implicit SDF field reconstruction. We select the top-k latent library codes that are have largest dot product with the predicted one and use Picasso to predict the object pose for each of them. Among these options we select the shape with lowest Chamfer loss. We show in the experiments that this improves performance compared to CRISP alone in all metrics. Note that we give the mean value of Chamfer losses across all objects for shape error following [46]. This motivates future work on shape correction. For example, a generative model could produce multiple plausible shape candidates and we can extend sampling to the shape hypotheses.

H. Physics-Guided Loss for Gradient Descent

Physics-guided losses are widely used in soft-constrained and physics-informed machine learning [35, 60]. Focusing on multi-object non-penetration, we develop a suite of loss terms used by CRISP-Syn+GD baseline in Section V-A. Recall that $\boldsymbol{T}_i \in \mathrm{SIM}(3)$ maps points in the camera frame to the object frame. Moreover, recall that $\Phi_i : \mathbb{R}^3 \to \mathbb{R}$ denotes the signed distance field (SDF) of object i in the local frame and Φ_{free} :

TABLE IX: Evaluation results on the e_{shape} ($\times 10^{-3}$ m) and e_{shape} (AUC %) metrics for the YCB-V dataset. Best.

Method	e_{shape} ↓	e_{shape} (AUC %) ↑		
		1 cm	2 cm	3 cm
CRISP-Syn+Picasso	26.95	47.6	63.3	77.4
CRISP-Syn+Picasso (Shape)	26.90	47.8	63.4	77.4

Fig. 10: IC-BIN dataset experiment. **Left**: RGB image. **Middle**: w/o object-free-space constraints. **Right**: w/ object-free-space constraints.

TABLE X: Evaluation results on the physics metrics, penetration metrics, observable correctness, and runtime on the YCB-V dataset. Best.

Method	SPS ↓	NPS ↓	OC ↑
CRISP-Syn+Picasso	8.71	3.99	55%
CRISP-Syn+Picasso (Shape)	8.64	3.74	56%

$\mathbb{R}^3 \to \mathbb{R}$ denotes the free-space SDF. We consider three types of penetration in this section:

a) Inter-object non-penetration:

$$\mathcal{L}_{\text{inter-obj}} = \frac{1}{M^W} \sum_{\boldsymbol{b} \in \mathcal{B}^W} \text{ReLU}\Big(\sum_{i \in [N]} \sigma\big(-\alpha\, \Phi_i(\boldsymbol{T}_i \cdot \boldsymbol{b})\big) - \tau \Big),$$

where $\sigma(z) = 1/(1+e^{-z})$ and $\mathcal{B}^W = \{\boldsymbol{b}_j\}_{j=1}^{M^W} \subset \mathbb{R}^3$ denote the camera frame object surface vertices in the overlapping axis-aligned bounding box regions across all objects. Empirically, we set $\alpha = 10000$, $\tau = 1.99$, $\lambda_{\text{io}} = 10$. Intuitively, if an object penetrates with another n objects, then $\sum_{i \in [N]} \sigma\big(-\alpha\, \Phi_i(\boldsymbol{T}_i \cdot \boldsymbol{b})\big)$ will be close to $n+1$.

b) Object-environment non-penetration:

$$\mathcal{L}_{\text{env-obj}} = \frac{1}{M^E} \sum_{\boldsymbol{b} \in \mathcal{B}^E} \sum_{i \in [N]} \text{ReLU}\Big(-\Phi_i(\boldsymbol{T}_i \cdot \boldsymbol{b})\Big),$$

where $\mathcal{B}^E = \{\boldsymbol{b}_j\}_{j=1}^{M^E} \subset \mathbb{R}^3$ denote the camera frame sampled points below the plane.

c) Object-free-space non-penetration:

$$\mathcal{L}_{\text{free}} = \frac{1}{M^F} \sum_{b \in \mathcal{B}^F} \text{ReLU}\left(e^{-\beta \Phi_{\text{env}}(\boldsymbol{b})} - 1\right),$$

where $\mathcal{B}^F = \{\boldsymbol{b}_j\}_{j=1}^{M^F} \subset \mathbb{R}^3$ denote the camera frame points lifted from pixel space and scaling factor $\beta = 10$. Depth noise can cause false positives, so we use an exponential penalty that focuses on large violations. The total physics-guided loss is

$$\mathcal{L}_{\text{phy}} = \lambda_{\text{inter-obj}} \mathcal{L}_{\text{inter-obj}} + \lambda_{\text{env-obj}} \mathcal{L}_{\text{env-obj}} + \lambda_{\text{free}} \mathcal{L}_{\text{free}},$$

with $\lambda_{\text{inter-obj}} = 10.0$, $\lambda_{\text{env-obj}} = 1.0$ and $\lambda_{\text{free}} = 10^{-10}$.

I. Failure Cases

a) Inaccurate Shape Prediction: As shown in Fig. 11, inaccurate SAM3D shape predictions for the two bottom Jenga blocks lead to errors in Picasso's pose estimation.

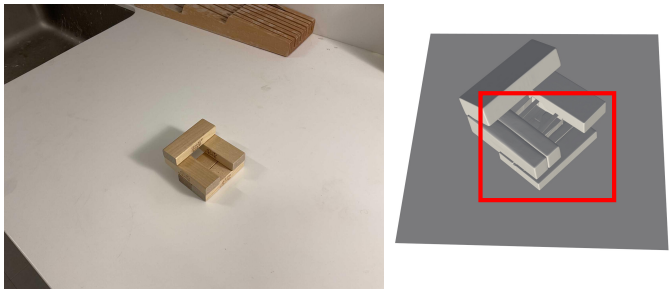


Fig. 11: A failure case due to inaccurate shape prediction.

Fig. 12: A failure case due to noisy and partial depth point clouds.

b) Noisy or Partial Depth Point Clouds: Noisy and partial depth point clouds also contribute to failures. In Fig. 12, the yellow box is only partially visible, yielding an incomplete depth point cloud. Picasso therefore selects an incorrect orientation that fits the observed points. Violations of the physics (free-space) constraints in the final output occur because the correct configuration is not in the top-16 candidates considered for constraint satisfaction due to depth noise.

c) Contact Scene Graph Approximation: We perform a single sweep over a scene graph approximated as a DAG, with the ground node as the root. In some cases, this approximation can lead to incorrect predictions. As shown in Fig. 13, under a bottom-up DAG construction, the holder is inferred before the bowl, so non-penetration constraints between them are not enforced when predicting the holder. Due to occlusion, this results in the holder being estimated upside down. In contrast, with a top-down construction, the bowl is inferred first and the holder second, activating the non-penetration constraints and yielding the correct prediction. This example motivates future work toward better inference schemes where upstream and downstream nodes in the contact scene graph mutually influence one another.

J. Human Study: Alignment with Physics Metrics

Setup. Our survey was performed between January 19th, 2026 and January 23rd, 2026. The survey involved two distinct groups: The first group was human participants from Amazon Mechanical Turk, which we refer to as the *Public* group. We launched two batches of surveys (30 participants each), collecting 60 total responses. After removing duplicate submissions (as Mechanical Turk workers may complete the same survey multiple times), we obtained 59 valid responses. The second group consisted of researchers spanning robotics and computer vision, which we refer to as the *Expert* group. To avoid bias toward any single research area, we included participants with expertise in 3D and low-level vision, control and

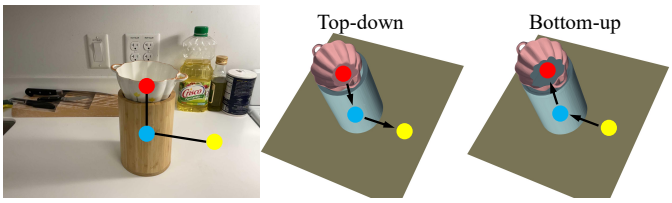


Fig. 13: Approximation of contact scene graph can lead to failure in downstream inference.

motion planning, reinforcement learning, multi-modal learning, underwater robotics, SLAM, and rehabilitation robotics.

We selected three frames from each YCB-V sequence —at the beginning, middle, and end— yielding 36 images in total. For every image, we ran both SAM3D and SAM3D+Picasso to obtain the reconstructions. For each reconstruction, we created a video showing orbit views of the reconstruction. Participants completed 36 trials. In each trial, they were shown two videos side by side: one generated by SAM3D and the other by SAM3D+Picasso. The video order was randomized per scene, and method names were anonymized to avoid bias. For both *Expert* and *Public* groups, participants received the following prompt: "*You will be given videos of a reconstructed scene. Imagine the scene is in simulation. Based on your intuition of the physical world —we know that if two objects penetrate they will be pushed away in opposite direction, if an object is floating in the air, it will fall— how would you evaluate the stability of the scene? Please evaluate from 1-7 where 1 means least stable and 7 means most stable.*"

Results. Fig. 14 compares human evaluation results between SAM3D and SAM3D+Picasso. We normalize both SPS and human evaluation scores, reversing the human evaluation metric so that lower values indicate better performance across all metrics. Both expert and public evaluators consistently rate SAM3D+Picasso as more physically plausible than SAM3D alone. While experts tend to assign lower scores overall (i.e., are more conservative in judging physical stability), the public is generally more optimistic. Moreover, the performance gap between SAM3D with and without the corrector is larger in the expert ratings, suggesting that experts are more sensitive to subtle physical inconsistencies. The performance gap between SAM3D with and without the corrector is even larger for SPS, indicating a better distinguishing capability of the metric. A detailed per sequence results of SPS, expert and public data are shown in Table XI. Overall, these results support that the corrector substantially improves the physical plausibility as for human intuition.

K. Hardware Demo

A hardware demo is conducted to illustrate how Picasso can be used to build a digital twin for object-relation reasoning. We set up three Jenga blocks supporting a marker, and the task is to remove one block without disturbing the marker. We use the Waveshare RoArm-M2-S robotic arm and manually define end-effector waypoints based on object pose estimation for the manipulation policy. For qualitative results, please see the attached video.

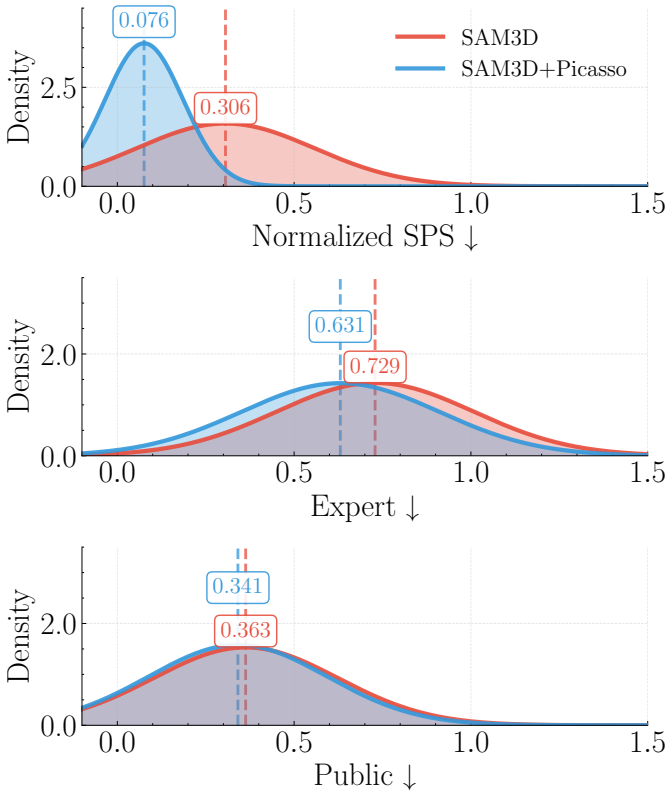


Fig. 14: Human evaluation and SPS comparison between SAM3D and SAM3D+Picasso. **Top:** SAM3D. **Bottom:** SAM3D+Picasso. For both *Experts* and *Public*, SAM3D+Picasso achieves higher physics plausibility.

TABLE XI: Human evaluation of physics plausibility and SPS across 12 YCB-Video trajectories. Human scores are on a 1-7 scale (higher is better) across 83 participants. SPS is averaged across 3 frames per trajectory. S: SAM3D. S+PC: SAM3D w/ physics-constrained corrector. Best.

Trajectory	SPS ↓		Expert ↑		Public ↑		Expert+Public ↑	
	S[9]	S+PC	S[9]	S+PC	S[9]	S+PC	S[9]	S+PC
traj_48	13.90	1.91	2.68	3.17	4.87	5.06	4.24	4.51
traj_49	3.05	0.54	2.32	2.88	4.76	4.88	4.04	4.29
traj_50	1.63	1.48	2.86	3.00	4.98	5.15	4.36	4.53
traj_51	7.75	0.65	2.78	3.03	5.07	4.94	4.41	4.38
traj_52	5.73	0.05	2.17	2.67	4.86	4.97	4.07	4.30
traj_53	7.66	0.00	2.39	3.74	4.69	4.99	4.01	4.62
traj_54	10.17	5.46	2.49	3.69	4.68	4.86	4.03	4.52
traj_55	6.20	2.49	2.21	3.72	4.61	5.07	3.90	4.68
traj_56	6.85	0.96	3.38	3.88	4.87	4.91	4.43	4.60
traj_57	2.08	0.06	2.44	3.53	4.78	4.97	4.10	4.54
traj_58	3.82	2.69	3.72	2.65	4.92	4.78	4.56	4.16
traj_59	4.58	1.84	2.08	2.65	4.80	4.87	4.01	4.22
Overall	6.12	1.51	2.63	3.22	4.82	4.95	4.18	4.45

L. Runtime

We report the average runtime of key modules on the Picasso dataset. Picasso takes 4.15s per object with the physics corrector and 1.97s per object without it. A single SAM3D query takes 7.66s on average, while a single Gemini query for constructing the contact scene graph takes 11.71s.