MessyKitchens: Contact-rich object-level 3D scene reconstruction

Junaid Ahmed Ansari[1*], Ran Ding[1*], Fabio Pizzati[1], and Ivan Laptev[1]

[1] Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE
{junaid.ansari,ran.ding,fabio.pizzati,ivan.laptev}@mbzuai.ac.ae

Abstract. Monocular 3D scene reconstruction has recently seen significant progress. Powered by the modern neural architectures and large-scale data, recent methods achieve high performance in depth estimation from a single image. Meanwhile, reconstructing and decomposing common scenes into individual 3D objects remains a hard challenge due to the large variety of objects, frequent occlusions and complex object relations. Notably, beyond shape and pose estimation of individual objects, applications in robotics and animation require physically-plausible scene reconstruction where objects obey physical principles of non-penetration and realistic contacts. In this work we advance object-level scene reconstruction along two directions. First, we introduce *MessyKitchens*, a new dataset with real-world scenes featuring cluttered environments and providing high-fidelity object-level ground truth in terms of 3D object shapes, poses and accurate object contacts. Second, we build on the recent SAM 3D approach for single-object reconstruction and extend it with Multi-Object Decoder (MOD) for joint object-level scene reconstruction. To validate our contributions, we demonstrate MessyKitchens to significantly improve previous datasets in registration accuracy and inter-object penetration. We also compare our multi-object reconstruction approach on three datasets and demonstrate consistent and significant improvements of MOD over the state of the art. Our new benchmark, code and pre-trained models are publicly available on our project website: https://messykitchens.github.io/.

Keywords: 3D reconstruction · Benchmark · Physical accuracy

1 Introduction

Accurate 3D scene reconstruction plays a pivotal role for many applications in digital arts and content creation, industrial inspection, surgery, heritage preservation, navigation as well as robot learning and simulation. While some tasks, e.g., navigation [2, 10, 43, 48], may only require free space estimation and scene-level surface reconstruction, other tasks, e.g., robotic manipulation [1, 7, 20, 37,

[*] Equal contribution.

Fig. 1: MessyKitchens benchmark. Images of real scenes and corresponding high-fidelity object-level 3D scenes reconstructions composed of accurate object scans.

39, 51, 56] and animation [6, 44, 55] often rely on detailed reconstruction of individual objects. Object-level scene reconstruction is a difficult task challenged by the large variety of object shapes and frequent occlusions. Moreover, beyond object shape and pose estimation, animation and simulation tasks require physically-plausible scene reconstruction with correct estimation of object contacts, deformations and other parameters.

Recent methods for 3D scene reconstruction have evolved from classic geometry-based formulations to learning-based approaches. The latter methods rely on the learned inductive biases and enable accurate shape predictions from a single image. In particular, several recent methods such as DepthAnything [60], VGGT [52] and Gen3C [46] significantly advance results for monocular depth estimation. In comparison, object-level scene reconstruction has received relatively less attention. Among existing methods, MIDI [28] and PartCrafter [38] present impressive results, but focus on synthetic scenes, while the recent SAM 3D [12] enables the estimation of the shape and pose of single objects in real images. Besides new methods, the progress in object-level scene reconstruction also requires realistic and high-fidelity benchmarks for training and evaluation. While several benchmarks exist [4, 20, 33], their 3D ground truth often suffers from the limited registration accuracy and inter-object penetrations.

To address these limitations, we advance object-level scene reconstruction on two fronts. *First*, we introduce *MessyKitchens*, a new dataset with 100 real-world scenes featuring cluttered environments along with high-fidelity object-level 3D ground truth. As shown in Figure 1, our dataset contains contact-rich scenes and 130 varying kitchen objects that have been scanned and registered with high precision. Moreover, MessyKitchens provides accurate object contacts, and hence enables evaluation of geometrically-precise and physically-realistic scene

reconstruction. In addition, we also provide a large-scale synthetic training set MessyKitchens-synthetic composed of 1.8k contact-rich scenes and 10.8k rendered images. *Second*, alongside the benchmark, we propose a method for joint object-level scene reconstruction. Our approach builds upon the recent SAM 3D framework for single-object reconstruction and extends it with a Multi-Object Decoder (MOD) that jointly predicts the geometry and poses of multiple objects in a scene. By reconstructing objects simultaneously, MOD captures contextual relationships and enforces more physically-plausible configurations.

In summary, we propose the following contributions:

- We introduce MessyKitchens, a new benchmark with cluttered real scenes and high-fidelity 3D object-level ground truth, including accurate object shapes, poses and contacts along with precise scene-level object registration.
- We propose Multi-Object Decoder (MOD), a method that extends SAM 3D to the joint modeling and reconstruction of multiple objects in a scene.
- Through extensive experiments we show MOD to outperform state-of-the-art object-level scene reconstruction in MessyKitchens, GraspNet-1B [20], HouseCat6D [33], and GraspClutter6D [4]. We also demonstrate significantly improved 3D accuracy of the MessyKitchens benchmark compared to other recent datasets.

2 Related Works

Existing benchmarks for 3D scenes. Early datasets such as LINEMOD [26], T-LESS [27], YCB-M [23], and YCB-Video [57] established standardized evaluation protocols for 3D pose estimation and object-level reconstruction, significantly advancing the field. However, they were captured in controlled laboratory settings and typically contain a limited number of objects and scenes with restricted category diversity. Datasets such as MP6D [11] and T-LESS [27] focus largely on industrial objects and lack representation of everyday kitchen categories, limiting their realism for domestic environments. Although synthetic datasets such as Falling Things [50] and ZeroGrasp-11B [30] scale up object count substantially, they remain purely simulation-based and do not reflect the challenges of real-world scene acquisition and annotation. More recent datasets, including GraspNet-1B [20], HouseCat6D [33], PhoCaL [54], Omni6DPose [65], KITchen [63], PACE [62], and GraspClutter6D [4], increase object diversity and incorporate kitchen and transparent objects. Nevertheless, they are often constrained by limited annotation accuracy, inaccurate contacts, insufficient scene complexity, or scalability (e.g., [33, 54] rely on robotic manipulator or motion capture systems for data collection). In contrast, MessyKitchens is collected using a portable 3D scanner enabling scene acquisition across diverse locations and achieves higher object-to-scene registration accuracy that naturally yields improved contacts, while remaining comparable in number and diversity of objects.

Object-centric 3D scene reconstruction. Object-centric 3D scene reconstruction has traditionally evolved from feed-forward and retrieval-based strategies. Feed-forward methods typically leverage encoder-decoder architectures to regress scene properties, such as geometry, instance labels, and poses, directly from images [14, 22, 40, 41, 53, 58, 64, 67], while retrieval-based approaches, such as DiffCAD [21] and ACDC [15], align high-quality 3D assets from existing databases to input [24, 31, 35, 36]. However, these methods are often limited by the scarcity of supervised 3D data and a heavy dependence on database diversity, which hinders their ability to generalize to complex out-of-distribution scenes [9, 16, 17]. Recently, compositional and modular generative paradigms [3, 13, 25, 49] have sought to model complex scenes utilizing large-scale perceptual and 3D priors [19, 32, 34, 45, 47, 66]. Frameworks such as MIDI [28] and PartCrafter [38] achieve high generative fidelity through multi-instance diffusion and Diffusion Transformers (DiT) [42]. However, these approaches are primarily developed on synthetic datasets and often face challenges when generalizing to the complexities of real-world captures. In contrast, SAM 3D [12] provides a highly scalable pipeline that delivers strong foundational results in diverse real-world scenes through robust data alignment. Yet, because SAM 3D typically processes objects as independent tokens, it lacks explicit, end-to-end reasoning regarding the spatial inter-dependencies between multiple objects. This independent treatment often leads to inaccuracies in the global spatial layout and imprecise relative object poses, posing a significant challenge for achieving spatially consistent 3D scene reconstructions in complex environments [61].

3 The MessyKitchens Benchmark

We introduce here our MessyKitchens benchmark. For it, our core aim is the evaluation of the accuracy of object-centered 3D scene reconstruction on challenging scenarios, using physically-curated ground truth data. We describe the process and the characteristics of the data in Section 3.1. In Section 3.2, we also provide a *synthetic* set, coined MessyKitchens-synthetic, to enable training on similar scenarios. Finally, in Section 4, we propose Multi-Object Decoder as a new simple baseline for 3D scene reconstruction.

3.1 Real data

Data acquisition. We now describe our data acquisition strategy. In total, we collect 100 real scenes, each composed of a variable number of kitchenware objects that depend on the difficulty level of the scene. We employ 130 objects in total, gathered from 10 different kitchens, for which we collect 3D scans with a Einstar Vega 3D scanner. We built a specific apparatus for scanning objects in isolation, displayed in Figure 2, left. In practice, we position the object on a transparent acrylic surface. Since the scanner does not detect the acrylic surface, this allows us to take multiple scans of the same object from different viewpoints without moving it. This allows us to greatly increase the precision

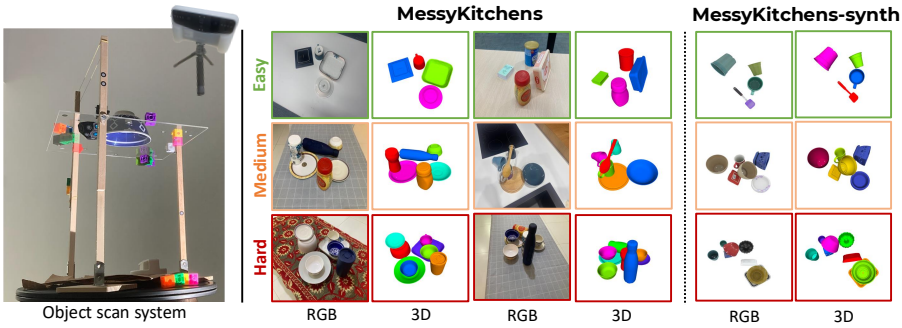


Fig. 2: On the left, we show our object scanning system. The transparent surface allows us to take multiple scans without moving the object. On the right, we show samples of MessyKitchens, for three difficulty levels. Scenes get more cluttered and with more sophisticated object interactions with the increase of the difficulty. We also provide a synthetic set (MessyKitchens-synthetic) usable for training, with constructed scenes similar to the real dataset.

of object scanning. We then collect two scans per object: a first scan collected from above the object, and a second from below, in order to capture the entire 3D geometry. These two scans are later aligned to obtain a dense 3D ground truth. To facilitate alignment, we position visual markers on the acrylic surface. We use double-sided reflective markers, positioned so that they can be seen by the scanner from either above or below the object. The full process takes $6 \sim 10$ minutes per object.

Difficulty levels. We then construct a scene by assembling pre-scanned objects on different surfaces, to promote variability. For scenes, we define three different difficulty levels depending on the configuration: **(1) Easy:** we include 4 objects, well separated and with minimal contact; **(2) Medium:** we include 6 objects, among which 4 are base objects lying on a flat surface, and the other 2 are stacked objects on top of the others, in equilibrium. We reduce the separation and impose more contacts with each other; **(3) Hard:** we use 8 objects, imposing maximum contact with each other. In addition to the characteristics of the medium setup, we include nested objects inserted one inside the other (such as a cup into a bowl). We display examples of our scenes configuration in Figure 2, center. After being assembled, the scenes are scanned with the same sensor that was used for objects. In particular, we carefully scan each constructed setup while ensuring stable tracking and complete surface coverage, so that even heavily occluded or contact-rich regions are captured as accurately as possible. The acquired scene scans are subsequently processed, cleaned, and decimated using the scanner software, preserving geometric fidelity (with a point-to-mesh error below 0.05 mm). We subsequently map textures to the mesh. In total, the scanning process takes up to 20 minutes per scene. The fully reconstructed scenes are exported for subsequent registration with the individual object models. Note that scenes are built incrementally: we first construct a hard scene and scan that

only. We perform on the hard scene the registration of existing objects. Then, we remove some objects from the hard scene and construct, consequently, a medium scene first, and finally an easy scene. This allows us to obtain the same configuration of some objects, with varying contacts. Note that the associated RGB to the scenes is instead varying, due to the different camera pose used for recording different difficulty levels. This could be useful for assessing the dependency of 3D reconstruction algorithms from camera viewpoints.

Registration. For registering the object models to their corresponding scenes, we first perform a manual coarse alignment which gives us the initial transformations for our registration pipeline. Our registration pipeline is in two stages: in the first stage, we randomly sample a set of 3D points from the surface of each object mesh, find the distance to closest on-surface point on the scene mesh, and then optimize the object transformations to minimize this cost, and in the second, taking the transformations from first stage as new initial estimates, we impose, at optimization time, that the normals of the scan and of the object are aligned for a given point. Indeed, most of our objects are thin and concave, so a point on the top surface and its opposite point on the bottom surface often have very similar distances to the scan. During automatic registration, the optimizer can therefore minimize error by placing the scan surface between the two walls of the object, incorrectly satisfying both sides. Penalizing for normals coherence can prevent this.

3.2 Synthetic data

Scene construction. To enable training on MessyKitchens, we construct a synthetic dataset with close similarity to our setup for real scene scanning. We call this set MessyKitchens-synthetic. We start by selecting a total of 42 kitchenware objects from the 3D assets of GSO [18]. We then generate scenes following the same difficulty levels as in our MessyKitchens real scenes. For easy scenes, we simply randomly place four objects on a flat surface. For medium scenes, after randomly placing four objects, we position two additional objects on top of existing ones, and we activate gravity to make the simulation physically realistic. We make sure that objects are effectively stable on top of others before including the scene inside the dataset. For hard scenes, we include stacked objects, as in the real setup. Ensuring a realistic randomized stacking of objects in synthetic simulation is nontrivial. To do so, we first compute object volume estimates, and drop appropriately sized objects into compatible support objects. All scenes are generated using concave, mesh-based collision, which is critical for achieving realistic stacked and nested interactions. In all cases, we ensure that our scenes are contact-rich and physically realistic. We visualize samples in Figure 2, right.

Rendering. Once we construct the 3D scene, we sample multiple views for enabling training of 3D reconstruction methods. To promote photorealism, we employ Blender's Cycles engine for the rendering of scenes and use the original texture files from the dataset.

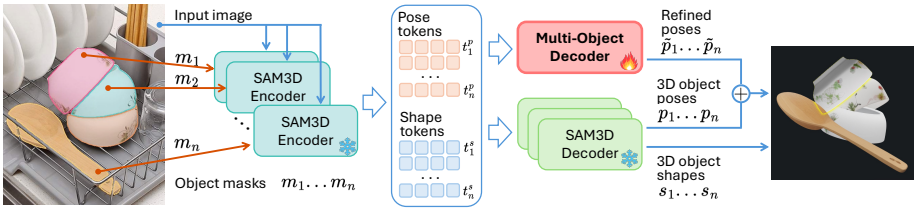


Fig. 3: Multi-Object Decoder for 3D reconstruction. SAM3D outputs 3D shapes from input images and masks. To impose scene-level constraints, we use a Multi-Object Decoder refining SAM3D prediction on the pose of the objects. The residual refined term is summed to the original prediction to obtain a scene-aware pose estimation.

We render 10 different views randominzing the camera angle, sampling from azimuth $\phi \in [0, 2\pi]$ and elevation $\iota \in [\pi/4, \pi/2]$. This sampling ensures diverse viewpoints while maintaining a top-down bias consistent with tabletop scenarios. Simultaneously with the image rendering, we extract instance-based semantic maps for all objects. For more details, please see supplementary material.

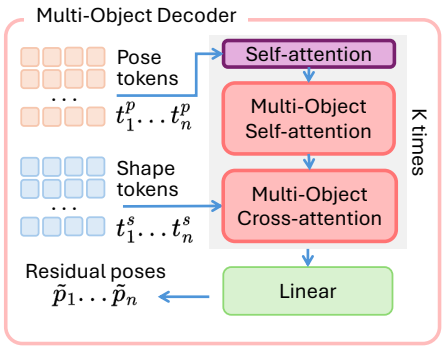


Fig. 4: Multi-Object Decoder. We inform pose tokens on scene-level context by using K blocks including multi-object self-attentions and cross-attention. We use pose and shape information from all objects to obtain residual pose correcting factors.

4 Multi-Object Decoder

4.1 Method details

To extract 3D objects from 2D images, SAM3D takes as input an image x and an object pixelwise mask m_i indicating the position of object i. It then returns a voxel-based shape prediction s and a 7-DOF pose $p = (\mathbf{q}, \mathbf{t}, \sigma)$ of the object in 3D space, where $\mathbf{q} \in \mathbb{H}_1$ is a unit quaternion representing 3D rotation, $\mathbf{t} \in \mathbb{R}^3$ is the translation vector, and $\sigma \in \mathbb{R}^+$ is the isotropic scaling factor. To do so, it processes x by extracting shape tokens $t^s \in \mathbb{R}^{F_s \times C}$ and pose tokens $t^p \in \mathbb{R}^{F_p \times C}$, used to estimate respectively s and p through the SAM3D decoder. F_s and F_p are the sequence lengths, and C the feature dimension. We refer to the original paper for details [12]. For a scene with N objects, we denote the aggregated scene tokens as $\mathbf{T}^s = \{t_1^s, \dots, t_N^s\} \in \mathbb{R}^{N \times F_s \times C}$ and $\mathbf{T}^p = \{t_1^p, \dots, t_N^p\} \in \mathbb{R}^{N \times F_p \times C}$, where t_i^s and t_i^p correspond to the shape and pose tokens of object i, respectively. Our objective is to train a Multi-Object Decoder to estimate residual scene-aware pose updates $\tilde{\mathbf{P}} = \{\tilde{p}_1, \dots, \tilde{p}_N\}$ from the aggregated tokens $\mathbf{T}^p$ and $\mathbf{T}^s$, and compute the final refined poses as $\mathbf{P} + \tilde{\mathbf{P}}$, encouraging globally consistent scene-level understanding. We visualize our approach in Figure 3. To correctly

predict scene-aware pose tokens, each pose prediction should be performed with awareness of both the pose and the shape of all other objects. We achieve this objective by building Multi-Object Decoder as a stack of K blocks composed of (i) a multi-object self-attention layer for pose tokens, which can correlate the pose of all objects, and (ii) a multi-object cross-attention that grounds the refined pose tokens to the shape tokens of all objects. As we show in Figure 4, in a block, we first refine pose tokens independently for each object by applying standard self-attention $\mathrm{SA}(\cdot)$ along the sequence dimension, and using N as a batch size:

$$\hat{\mathbf{T}}^p = \mathrm{SA}(\mathbf{T}^p), \qquad \hat{\mathbf{T}}^p \in \mathbb{R}^{N \times F_p \times C}. \tag{1}$$

This step increases the expressive capacity of pose features while preserving object-wise separation. We then enable cross-object reasoning by flattening the object and token dimensions of $\mathbf{T}^p$ into a single sequence of length NF_p, similarly to related literature [28, 38], hence $\hat{\mathbf{T}}^p \in \mathbb{R}^{1 \times (NF_p) \times C}$ now. We then process it with a self-attention layer $\mathrm{SA}_{\mathrm{multi}}(\cdot)$:

$$\tilde{\mathbf{T}}^p = \mathrm{SA}_{\mathrm{multi}}(\hat{\mathbf{T}}^p), \qquad \tilde{\mathbf{T}}^p \in \mathbb{R}^{1 \times (NF_p) \times C} \tag{2}$$

Finally, to ground pose predictions to geometry, we similarly process shape and pose tokens with a multi-object cross-attention, using the aggregated pose tokens $\tilde{\mathbf{T}}^p$ as queries and reshaped shape tokens $\mathbf{T}^s \in \mathbb{R}^{1 \times (NF_s) \times C}$ as keys and values:

$$\mathbf{T}_{\mathrm{out}} = \mathrm{CA}_{\mathrm{multi}}(\tilde{\mathbf{T}}^p, \mathbf{T}^s), \qquad \mathbf{T}_{\mathrm{out}} \in \mathbb{R}^{1 \times (NF_p) \times C}. \tag{3}$$

The result $\mathbf{T}_{\mathrm{out}}$ is finally reshaped to $\mathbb{R}^{N \times F_p \times C}$, and is provided as input for the next block. The output of the last block is decoded to $\tilde{\mathbf{P}}$ with a linear layer, independently for each refined pose.

4.2 Training and inference

We train Multi-Object Decoder with a weighted combination of rotation, translation, and scale losses. First, we impose a geometry term $\mathcal{L}_{\mathrm{CD}}$ based on the Chamfer distance (CD) between the predicted and GT shapes. Then, we represent rotations as unit quaternions $\mathbf{q} \in \mathbb{H}_1$ and take advantage of an alignment term based on the predicted and ground truth quaternion rotation $\hat{\mathbf{q}}$: $\mathcal{L}_{\mathrm{ip}} = 1 - \langle \mathbf{q}, \hat{\mathbf{q}} \rangle^2$. This loss accounts for the double-cover property of the quaternions, where $\mathbf{q}$ and $-\mathbf{q}$ represent the same physical rotation; by squaring the inner product, we effectively minimize the geodesic distance on the $SO(3)$ manifold without sign ambiguity [29]. Translation and scale are supervised with standard regression losses between outputs (the translation vector $\mathbf{t} \in \mathbb{R}^3$ and the isotropic scaling factor $\sigma \in \mathbb{R}^+$) and the ground truth, *i.e.* $\mathcal{L}_t$ and $\mathcal{L}_s$, respectively. Explicitly, the symmetric Chamfer distance is:

$$\mathcal{L}_{\mathrm{CD}}(S, \hat{S}) = \frac{1}{|S|} \sum_{x \in S} \min_{\hat{x} \in \hat{S}} \|x - \hat{x}\|_2^2 + \frac{1}{|\hat{S}|} \sum_{\hat{x} \in \hat{S}} \min_{x \in S} \|x - \hat{x}\|_2^2. \tag{4}$$

This geometric term is vital for handling object symmetry; since it relies on nearest-neighbor correspondence rather than fixed point-wise mapping, geometrically identical views of symmetric objects result in an equivalent loss, preventing the model from being penalized for valid but non-unique orientations. Considering that our training data and SAM 3D outputs may lie in different canonical spaces, to obtain reliable ground truth, we first estimate a global $Sim(3)$ transformation between the predicted and ground truth scenes using ICP [5]. We then match predicted objects to ground truth ones and, for each matched pair, refine the object pose by performing $Sim(3)$ ICP from the SAM 3D object to its ground truth counterpart. Finally, we decompose the resulting $Sim(3)$ transformation into rotation, translation, and scale (denoted as $\hat{\mathbf{q}}, \hat{\mathbf{t}}$, and $\hat{\sigma}$ respectively) to serve as direct supervision. Our final objective is:

$$\mathcal{L} = 0.1\,\mathcal{L}_{\text{CD}} + 100\,\mathcal{L}_t + 100\,\mathcal{L}_s + 10\,\mathcal{L}_{\text{ip}}. \tag{5}$$

At inference, we extract tokens for each object detected by SAM 3 [8] and refine their pose predictions with Multi-Object Decoder. The structural outputs of SAM 3D, such as voxels or meshes, remain unchanged but are accurately repositioned in the 3D scene to ensure global coherence and physical consistency.

5 Experiments

5.1 Setup and baselines

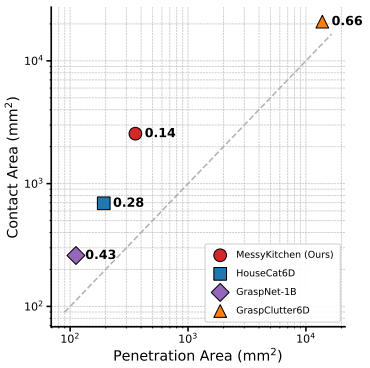
Datasets We compare with multiple concurrent benchmarks including household and kitchenware objects, such as T-LESS [27], LINEMOD [26], YCB-Video [57], MP6D [11], GraspNet-1B [20], GraspClutter6D [4]. For the evaluation of MOD, besides MessyKitchens, we use GraspNet-1B [20], HouseCat6D [33], and GraspClutter6D [4] as an out-of-distribution test set.

Methods We compare the 3D reconstruction of Multi-Object Decoder against three major object-level baselines, PartCrafter [38], MIDI [28], and SAM 3D [12]. PartCrafter is the only method that does not require a segmentation map of the objects as input. For all the others and ours, we use the same segmentation map extracted by SAM 3 [8]. For all, we compare in terms of intersection-over-union (IoU) and Chamfer Distance (CD). For both metrics, we report object-level and scene-level values.

Training setup We train our Multi-Object Decoder on 4 NVIDIA A100 GPUs (40GB). For training, we use MessyKitchens-synthetic, sampling 600 scenes per difficulty level (easy/medium/hard), and rendering 6 images per scene, for a total of 10800 images. MOD adds a few parameters to SAM 3D (approximately 81 million), and training for 10 epochs takes approximately 2 hours. We set $K = 3$ for the architecture of MOD. We used a learning rate of 5×10^{-5} with a linear warm-up during the first 10% of training.

Dataset	Sensor	$\mu_{\|\delta\|}$	$\text{med}_{\|\delta\|}$	σ_δ
T-LESS [27]	Camarine	4.28	2.46	7.72
	Kinect v2	8.40	5.45	11.36
LINEMOD [26]	Kinect v2	5.89	5.57	1.47
YCB-Video [57]	Xtion Pro Live	3.95	3.66	2.26
MP6D [11]	Tuyang FM851-E2	3.54	2.70	**0.17**
GraspNet-1B [20]	RealSense D435	7.69	4.95	14.30
	Azure Kinect	14.79	9.54	20.20
GraspClutter6D [4]	Zivid	3.22	1.55	11.10
	RealSense D415	5.71	3.77	14.67
	RealSense D435	7.02	4.58	13.82
	Azure Kinect	13.85	6.83	32.59
MessyKitchens	Einstar Vega	**1.62**	**0.91**	3.83

(a) Registration accuracy



(b) Contacts and penetration

Fig. 5: Comparison with other benchmarks. In Table a, we show that MessyKitchens yields significant improvements in registration accuracy, measured with depth errors (mm), compared to others. In Figure b, we calculate the ratio between penetration area and contacts surface area. MessyKitchens exhibits the best ratio, demonstrating the high quality of our cluttered scenes, resulting in physically-realistic contacts.

5.2 Data quality

Registration accuracy Registration accuracy is crucial for faithful 3D reconstruction. Following [4, 27], we assess registration quality by measuring the depth discrepancy between the rendered depth of registered objects and the ground-truth scan depth. In Figure 5a, we report the mean ($\mu_{|\delta|}$) and median ($\text{med}_{|\delta|}$) absolute depth error, together with the standard deviation σ_δ, all in millimeters. As shown, MessyKitchens achieves *very accurate registration*, with a mean error of **1.62** mm, corresponding to a 49.7% relative improvement over the second-best benchmark, GraspClutter6D (3.22). We observe a similar trend for the median error, where MessyKitchens attains **0.91** mm, improving by 41.3% over the second best (1.55). These results highlight the precision of our data and the effectiveness of our automatic registration pipeline, demonstrating that MessyKitchens provides a reliable foundation for object-level 3D reconstruction.

Table 1: Contacts across difficulty levels. We report contacts accuracy metrics across difficulty levels. The ratio between penetration and contacts in medium and hard scenes is similar, showcasing the quality of our data.

Split	**Contacts measures** (mm^2)		
	C. Area	P. Area	Ratio
Easy	14.66	1.062	0.0724
Medium	2593	397.4	0.1533
Hard	5892	793.5	0.1347

Contacts and penetrations Realistic modeling of contacts is essential in cluttered scenes, as many downstream tasks rely on physically plausible object interactions. However, imprecise dataset construction may introduce unrealistic

Table 2: Effectiveness of Multi-Object Decoder. On MessyKitchens, GraspNet-1B, HouseCat6D, and GraspClutter6D, SAM 3D+MOD achieves state-of-the-art object-level 3D scene reconstruction. We measure both object-level metrics and scene-level ones. GraspClutter6D [4] is a challenging out-of-distribution benchmark with high clutter; all MOD evaluations are zero-shot.

		MessyKitchens		GraspNet-1B		HouseCat6D		GraspClutter6D	
	Method	IoU↑	CD↓	IoU↑	CD↓	IoU↑	CD↓	IoU↑	CD↓
Object	PartCrafter	0.071	0.495	0.020	0.956	0.029	0.852	0.067	0.674
	MIDI	0.186	0.285	0.067	0.640	0.092	0.433	0.086	0.500
	SAM 3D	0.409	0.064	0.336	0.082	0.325	0.125	0.328	0.105
	MOD	**0.445**	**0.061**	**0.344**	**0.078**	**0.404**	**0.100**	**0.340**	**0.103**
Scene	PartCrafter	0.133	0.228	0.075	0.355	0.114	0.289	0.227	0.227
	MIDI	0.238	0.165	0.121	0.327	0.172	0.215	0.213	0.201
	SAM 3D	0.431	0.054	0.356	0.074	0.374	0.099	0.487	0.059
	MOD	**0.472**	**0.050**	**0.377**	**0.069**	**0.458**	**0.079**	**0.496**	**0.058**

inter-object penetrations, limiting the reliability of the data for contact reasoning and physics-based applications. In Figure 5b, we compare the datasets by jointly analyzing the contact and penetration statistics. On the y-axis, we report the contact area, computed as the surface of points lying within 2.5mm between distinct objects. On the x-axis, we measure the penetration area, defined as the area of surfaces from one object intersecting another, detected via voxelization and intersection testing. Next to each point, we report the ratio between penetration surface and contact surface areas. While GraspClutter6D exhibits large contact regions, it also shows substantial penetration, resulting in an unfavorable ratio (0.66), suggesting that many contacts in the 3D scenes are not physically realistic. In contrast, MessyKitchens achieves the best ratio (**0.14**), showcasing the precision of our registered cluttered scenes. This serves as a further proof of the physical consistency of object interactions in MessyKitchens. Finally, in Table 1, we report ratios between contacts and penetration in easy/medium/hard scenes. We notice that the ratio obtained in medium/hard scenes is similar, highlighting that our registration is robust enough for physically-accurate contacts independently from the complexity. Easy scenes have no contacts.

5.3 3D object-based reconstruction

Effectiveness of MOD In Table 2, we compare Multi-Object Decoder against state-of-the-art baselines for object-centric 3D reconstruction. We train Multi-Object Decoder exclusively on MessyKitchens-Synthetic, while evaluating SAM 3D, PartCrafter, and MIDI in a zero-shot setting. Evaluation is conducted on MessyKitchens together with three out-of-distribution benchmarks, GraspNet-1B [20], HouseCat6D [33], and GraspClutter6D [4], on which all Multi-Object Decoder evaluations are zero-shot. As predicted 3D reconstructions and ground-truth scenes may not share a common coordinate frame, we perform scene-level alignment using Sim(3) ICP. To mitigate sensitivity to initialization and avoid

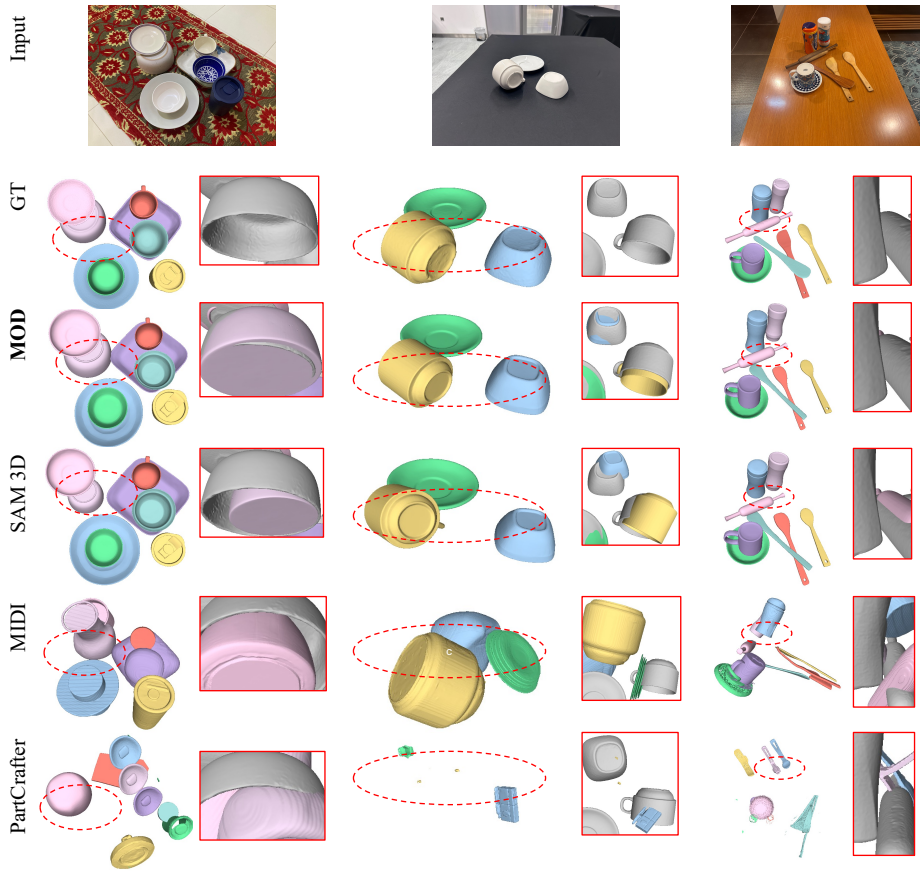


Fig. 6: Qualitative comparison. We show examples of 3D reconstructions for MOD and alternative methods, on MessyKitchens. In the insets, we show significant scene-level improvements over SAM 3D, demonstrating the effectiveness of MOD. The gray shapes are ground truth.

alignment bias, we run ICP three times and report the best result. Naive inference with SAM 3D already substantially outperforms competing methods across all benchmarks, achieving, for example, a +19.3% improvement over the second best (MIDI) in scene-level reconstruction (0.238 vs 0.431), motivating our design choice to use Multi-Object Decoder as a plug-in component to SAM 3D. However, incorporating Multi-Object Decoder further boosts performance consistently. For instance, for object-level IoU, training solely on synthetic data yields improvements on MessyKitchens (0.409 vs **0.445** with Multi-Object Decoder), as well as gains on GraspNet-1B (0.336 vs **0.344**), HouseCat6D (0.325 vs **0.404**), and GraspClutter6D (0.328 vs **0.340**). These results indicate that MessyKitchens provides effective supervision for robust object-based 3D reconstruction and highlight the importance of accurate contacts and object pose/scale refinement, as even minor geometric corrections lead to notable improvements.

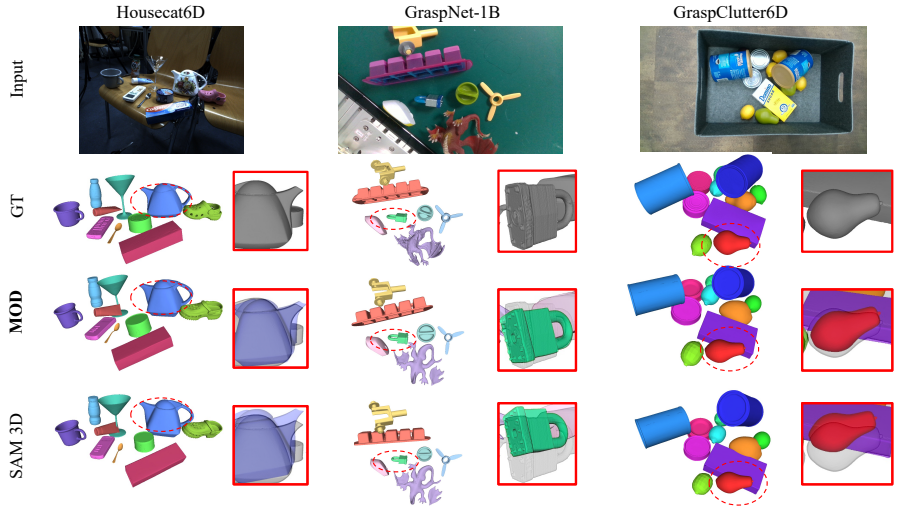


Fig. 7: Qualitative comparison across Housecat6D, GraspNet-1B, and GraspClutter6D. We visualize 3D scene reconstructions for MOD and the SAM 3D baseline. Our method consistently produces physically more plausible results with fewer interpenetration and more accurate grounding compared to the baseline. Ground-truth meshes are overlaid in gray with transparency for reference.

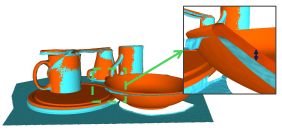
Furthermore, Multi-Object Decoder demonstrates strong generalization under domain shift: HouseCat6D, GraspNet-1B, and GraspClutter6D contain object categories distinct from kitchenware, confirming OOD generalization to different objects.

Finally, in Table 3 we quantify the physical plausibility of the reconstructions on MK using the contact metrics of PhySIC [59], built from a graph linking objects in contact. MOD ranks first, showing that our contribution improves contact quality.

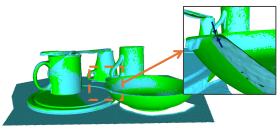
Table 3: Contact accuracy. We build contact graphs as [59] to evaluate the quality of MOD-reconstructed contacts.

Method	Contact (%)		
	Prec↑	Rec↑	F1↑
MIDI	38.9	48.1	43.0
PartCrafter	27.2	17.4	21.2
SAM 3D	66.0	48.3	55.8
MOD	**70.0**	**62.5**	**66.1**

Qualitative results We report in Figure 6 a qualitative comparison between Multi-Object Decoder and alternative methods on MessyKitchens. As shown, PartCrafter and MIDI struggle to generalize to the evaluated setups, often producing incomplete or geometrically inconsistent reconstructions. SAM 3D instead generates visually realistic object shapes across MessyKitchens and the other datasets. Multi-Object Decoder builds on these predictions by refining only the pose and scale of the detected objects, leaving their geometry unchanged. In the insets, we present close-up comparisons between SAM 3D and Multi-Object Decoder outputs (ground truth shown as transparent gray surfaces). The addition of Multi-Object Decoder consistently improves alignment in the presence of object interactions, correcting pose and



(a) Distance only



(b) Distance+normals

Registration	$\mu_{\lvert\delta\rvert}$	$\text{med}_{\lvert\delta\rvert}$	σ_δ
Manual	4.689	3.209	7.439
Distance only	2.892	2.141	4.822
Distance+normals	**1.615**	**0.911**	**3.827**

(c) Quantitative

Fig. 8: Registration ablation. We show that our approach based on distance+normal (Fig. b) registration considerably improves results with respect distance only (Fig. a). Metrics on registration (Table c) agree with our evaluation.

scale estimates by enforcing scene-level geometric consistency and more accurate inter-object spatial relationships. In particular, the last column shows how Multi-Object Decoder correctly reconstructs two objects in contact with each other, thanks to our precise contact-rich training data in MessyKitchens. We further report qualitative reconstructions on the three out-of-distribution (OOD) benchmarks in Figure 7: GraspNet-1B, HouseCat6D, and GraspClutter6D. As shown, Multi-Object Decoder demonstrates superior generalization compared to SAM 3D. Despite the significant domain gap between our synthetic training data and these real-world captures, Multi-Object Decoder maintains global scene coherence, resolving common artifacts such as inter-object penetrations and unstable "floating" poses that frequently occur in the baseline reconstructions. This highlights the effectiveness of our learned object-level priors in reasoning about complex, contact-rich interactions across diverse environments.

5.4 Ablation studies

Registration strategy We ablate our design for the registration strategy, following the same metrics reported in Figure 5a. First, we show the rough manual initialization used for our registration. Then, we perform our automatic registration, with a naive distance criteria or our matching criteria based on normals described in Sec. 3.1. From results in Figure 8, we outperform considerably the alternatives. This shows the effectiveness of our automatic registration pipeline.

Multi-object attention In Multi-Object Decoder, we use multi-object self- and cross- attention, for pose and shape tokens. Now, we investigate the effectiveness of our choice. We compare our design (S+P) using both multi-object attention in shape (S) and pose (P) tokens with alternative designs in which we use multi-object attention for shape or pose only. In these, we replace the multi-object attention with a standard cross- or self-attention, operating on a single object. In Table 4a, we show that our setup achieves the best results, suggesting that the interaction between pose and shape tokens for all objects is important.

Architecture We propose an ablation on the blocks used for the Multi-Object Decoder. In Table 4b, we variate K, the number of blocks, in the range $K = \{1, 3, 6\}$. From our results, $K = 3$ yields the best results, while the addition of more blocks (K=6) tends to degrade results. We hypothesize that this is due to the richness of the features of SAM 3D, where we plug MOD on. Indeed, it

Table 4: MOD ablation studies. In Table a, we investigate the effects of the multi-object attention layers, showing that both scene-level information about pose and shape contribute to best performance. In Table b, we report that 3 transformer blocks to build MOD ($K = 3$) yield the best results.

		MessyKitchens	
	MOD Setup	IoU↑	CD↓
Object	Shape only	0.438	0.063
	Pose only	0.432	0.065
	S+P (Ours)	**0.445**	**0.061**
Scene	Shape only	0.463	0.054
	Pose only	0.457	0.056
	S+P (Ours)	**0.472**	**0.050**

(a) Multi-object attention

		MessyKitchens	
	Blocks Setup	IoU↑	CD↓
Object	$K = 1$	0.441	0.061
	$K = 3$ **(Ours)**	**0.445**	**0.061**
	$K = 6$	0.403	0.065
Scene	$K = 1$	0.467	0.051
	$K = 3$ **(Ours)**	**0.472**	**0.050**
	$K = 6$	0.427	0.059

(b) Number of blocks

may be that SAM 3D does not necessitate many non-linearities and complex architectures to encode scene awareness, benefiting from a smaller MOD.

6 Conclusion

In this work, we addressed the challenging problem of physically-plausible, object-level 3D scene reconstruction from single images. To overcome the limitations of existing datasets, we introduced MessyKitchens, a novel benchmark featuring cluttered, contact-rich real-world scenes. By utilizing a rigorous data acquisition and normals-aware registration pipeline, MessyKitchens provides high-fidelity 3D ground truth with significantly lower inter-object penetration compared to previous datasets, setting a new standard for evaluating physical consistency. Furthermore, we proposed the MOD (Multi-Object Decoder), a simple yet highly effective extension for the single-object SAM 3D framework that combines several multi-object attentions. Extensive experiments demonstrate that our approach significantly outperforms state-of-the-art baselines on MessyKitchens, GraspNet-1B, HouseCat6D, and GraspClutter6D showing strong out-of-distribution generalization and high-quality results. Ultimately, we believe that MessyKitchens and MOD will provide a robust foundation for future research in physics-consistent 3D computer vision, facilitating advancements in downstream applications such as robotic manipulation, virtual reality, and 3D animation.

Acknowledgment

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean Government (MSIT) (No. RS-2024-00457882, National AI Research Lab Project).

We would also like to thank Komal Kumar, Mohamad Kassab, Gustavo Stahl, Ramazan Fazylov, David Romero, MBZUAI Cafe and Canteen, Amog Rao, Dinura Dissanayake and Mingfei Han for helping us with their objects.

References

1. Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., et al.: Do as i can, not as i say: Grounding language in robotic affordances. CoRL (2022) 1
2. Ansari, J.A., Tourani, S., Kumar, G., Bhowmick, B.: Exploring social motion latent space and human awareness for effective robot navigation in crowded environments. In: IROS. IEEE 1
3. Ardelean, A., Özer, M., Egger, B.: Gen3dsr: Generalizable 3d scene reconstruction via divide and conquer from a single view. In: 2025 International Conference on 3D Vision (3DV). pp. 616–626. IEEE (2025) 4
4. Back, S., Lee, J., Kim, K., Rho, H., Lee, G., Kang, R., Lee, S., Noh, S., Lee, Y., Lee, T., et al.: Graspclutter6d: A large-scale real-world dataset for robust perception and grasping in cluttered scenes. RA-L (2025) 2, 3, 9, 10, 11
5. Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In: Sensor fusion IV: control paradigms and data structures. Spie (1992) 9
6. Blender Online Community: Blender - a 3d modelling and rendering package (2024), http://www.blender.org 2
7. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al.: Rt-1: Robotics transformer for real-world control at scale. RSS (2023) 1
8. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. arxiv (2025) 9
9. Chang, A.X., et al.: Shapenet: An information-rich 3d model repository. arXiv (2015) 4
10. Chaplot, D.S., Gandhi, D., Gupta, S., Gupta, A., Salakhutdinov, R.: Learning to explore using active neural slam. ICLR (2020) 1
11. Chen, L., Yang, H., Wu, C., Wu, S.: Mp6d: An rgb-d dataset for metal parts' 6d pose estimation. RA-L (2022) 3, 9, 10
12. Chen, X., Chu, F.J., Gleize, P., Liang, K.J., Sax, A., Tang, H., Wang, W., Guo, M., Hardin, T., Li, X., et al.: Sam 3d: 3dfy anything in images. arXiv (2025) 2, 4, 7, 9
13. Chen, Y., Wang, T., Wu, T., Pan, X., Jia, K., Liu, Z.: Comboverse: Compositional 3d assets creation using spatially-aware diffusion guidance. In: ECCV (2024) 4
14. Dahnert, M., Hou, J., Nießner, M., Dai, A.: Panoptic 3d scene reconstruction from a single rgb image. NeurIPS (2021) 4
15. Dai, T., Wong, J., Jiang, Y., Wang, C., Gokmen, C., Zhang, R., Wu, J., Fei-Fei, L.: Automated creation of digital cousins for robust policy learning. arXiv preprint arXiv:2410.07408 (2024) 4
16. Deitke, M., et al.: Objaverse: A universe of annotated 3d objects. CVPR (2023) 4
17. Deitke, M., et al.: Objaverse-xl: A universe of 10m+ 3d objects. NeurIPS (2024) 4
18. Downs, L., Francis, A., Koenig, N., Kinman, B., Hickman, R., Reymann, K., McHugh, T.B., Vanhoucke, V.: Google scanned objects: A high-quality dataset of 3d scanned household items. In: ICRA (2022) 6
19. Eftekhar, A., Sax, A., Malik, J., Zamir, A.: Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In: ICCV (2021) 4
20. Fang, H.S., Wang, C., Gou, M., Lu, C.: Graspnet-1billion: A large-scale benchmark for general object grasping. In: CVPR (2020) 1, 2, 3, 9, 10, 11

21. Gao, D., Rozenberszki, D., Leutenegger, S., Dai, A.: Diffcad: Weakly-supervised probabilistic cad model retrieval and alignment from an rgb image. TOG (2024) 4
22. Gkioxari, G., Malik, J., Johnson, J.: Mesh r-cnn. In: ICCV (2019) 4
23. Grenzdörffer, T., Günther, M., Hertzberg, J.: Ycb-m: A multi-camera rgb-d dataset for object recognition and 6dof pose estimation. In: ICRA (2020) 3
24. Gümeli, C., Dai, A., Nießner, M.: Roca: Robust cad model retrieval and alignment from a single image. In: CVPR (2022) 4
25. Han, H., et al.: Reparo: Compositional 3d assets generation with differentiable 3d layout alignment. ICCV (2025) 4
26. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., Navab, N.: Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In: ACCV (2012) 3, 9, 10
27. Hodan, T., Haluza, P., Obdržálek, Š., Matas, J., Lourakis, M., Zabulis, X.: T-less: An rgb-d dataset for 6d pose estimation of texture-less objects. In: WACV (2017) 3, 9, 10
28. Huang, Z., Guo, Y.C., An, X., Yang, Y., Li, Y., Zou, Z.X., Liang, D., Liu, X., Cao, Y.P., Sheng, L.: Midi: Multi-instance diffusion for single image to 3d scene generation. In: CVPR (2025) 2, 4, 8, 9
29. Huynh, D.Q.: Metrics for 3d rotations: Comparison and analysis. Journal of Mathematical Imaging and Vision (2009) 8
30. Iwase, S., Irshad, M.Z., Liu, K., Guizilini, V., Lee, R., Ikeda, T., Amma, A., Nishiwaki, K., Kitani, K., Ambrus, R., et al.: Zerograsp: Zero-shot shape reconstruction enabled robotic grasping. In: CVPR (2025) 3
31. Izadinia, H., Shan, Q., Seitz, S.M.: Im2cad. In: CVPR (2017) 4
32. Jun, H., Nichol, A.: Shap-e: Generating conditional 3d implicit functions. arXiv preprint arXiv:2305.02463 (2023) 4
33. Jung, H., Wu, S.C., Ruhkamp, P., Zhai, G., Schieber, H., Rizzoli, G., Wang, P., Zhao, H., Garattoni, L., Meier, S., et al.: Housecat6d-a large-scale multi-modal category level 6d object perception dataset with household objects in realistic scenarios. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22498–22508 (2024) 2, 3, 9, 11
34. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. In: ICCV (2023) 4
35. Kuo, W., Angelova, A., Lin, T.Y., Dai, A.: Mask2cad: 3d shape prediction by learning to segment and retrieve. In: ECCV (2020) 4
36. Kuo, W., Angelova, A., Lin, T.Y., Dai, A.: Patch2cad: Patchwise embedding learning for in-the-wild shape retrieval from a single image. In: ICCV (2021) 4
37. Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., Quillen, D.: Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. The International journal of robotics research (2018) 1
38. Lin, Y., Lin, C., Pan, P., Yan, H., Feng, Y., Mu, Y., Fragkiadaki, K.: Partcrafter: Structured 3d mesh generation via compositional latent diffusion transformers. In: NeurIPS (2025) 2, 4, 8, 9
39. Mousavian, A., Eppner, C., Fox, D.: 6-dof graspnet: Variational grasp generation for object manipulation. In: ICCV (2019) 1
40. Nie, Y., Han, X., Guo, S., Zheng, Y., Chang, J., Zhang, J.J.: Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In: CVPR (2020) 4

41. Paschalidou, D., Kar, A., Shugrina, M., Kreis, K., Geiger, A., Fidler, S.: Atiss: Autoregressive transformers for indoor scene synthesis. Advances in Neural Information Processing Systems (NeurIPS) **34**, 12013–12026 (2021) 4
42. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV (2023) 4
43. Philion, J., Fidler, S.: Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In: ECCV (2020) 1
44. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-nerf: Neural radiance fields for dynamic scenes. In: CVPR (2021) 2
45. Ren, T., et al.: Grounded sam: Assembling open-world models for diverse visual tasks. arXiv (2024) 4
46. Ren, X., Shen, T., Huang, J., Ling, H., Lu, Y., Nimier-David, M., Müller, T., Keller, A., Fidler, S., Gao, J.: GEN3C: 3D-Informed world-consistent video generation with precise camera control. In: CVPR (2025) 2
47. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022) 4
48. Shah, D., Eysenbach, B., Kahn, G., Rhinehart, N., Levine, S.: Ving: Learning open-world navigation with visual goals. In: ICRA (2021) 1
49. Tang, J., Nie, Y., Markhasin, L., Dai, A., Thies, J., Nießner, M.: Diffuscene: Denoising diffusion models for generative indoor scene synthesis. CVPR (2024) 4
50. Tremblay, J., To, T., Birchfield, S.: Falling things: A synthetic dataset for 3d object detection and pose estimation. In: CVPR Workshops (2018) 3
51. Wang, H., Shahriar, F., Azimi, A., Vasan, G., Mahmood, R., Bellinger, C.: Versatile and generalizable manipulation via goal-conditioned reinforcement learning with grounded object detection. arXiv (2025) 1
52. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: VGGT: Visual geometry grounded transformer. In: CVPR (2025) 2
53. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2mesh: Generating 3d mesh models from single rgb images. In: ECCV (2018) 4
54. Wang, P., Jung, H., Li, Y., Shen, S., Srikanth, R.P., Garattoni, L., Meier, S., Navab, N., Busam, B.: Phocal: A multi-modal dataset for category-level object pose estimation with photometrically challenging objects. In: CVPR (2022) 3
55. Wu, Z., Yu, C., Wang, F., Bai, X.: Animateanymesh: A feed-forward 4d foundation model for text-driven universal mesh animation. In: ICCV (2025) 2
56. Xia, Y., Ding, R., Qin, Z., Zhan, G., Zhou, K., Yang, L., Dong, H., Cremers, D.: Targo: benchmarking target-driven object grasping under occlusions. arXiv preprint arXiv:2407.06168 (2024) 1
57. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. RSS (2017) 3, 9, 10
58. Xie, H., Yao, H., Sun, X., Zhou, S., Zhang, S.: Pix2vox: Context-aware 3D reconstruction from single and multi-view images. In: ICCV (2019) 4
59. Yalandur Muralidhar, P., Xue, Y., Xie, X., Kostyrko, M., Pons-Moll, G.: Physic: Physically plausible 3d human-scene interaction and contact from a single image. In: SIGGRAPH Asia 2025 Conference Papers (2025) 13
60. Yang, L., Kang, B., Huang, Z., Xu, X., Feng, J., Zhao, H.: Depth Anything: Uleashing the power of large-scale unlabeled data. In: CVPR (2024) 2
61. Yao, K., Zhang, L., Yan, X., Zeng, Y., Zhang, Q., Xu, L., Yang, W., Gu, J., Yu, J.: Cast: Component-aligned 3d scene reconstruction from an rgb image. ACM Transactions on Graphics (TOG) **44**(4), 1–19 (2025) 4
62. You, Y., Xiong, K., Yang, Z., Huang, Z., Zhou, J., Shi, R., Fang, Z., Harley, A.W., Guibas, L., Lu, C.: Pace: Pose annotations in cluttered environments (2024) 3

63. Younes, A., Asfour, T.: Kitchen: A real-world benchmark and dataset for 6d object pose estimation in kitchen environments. In: Humanoids (2024) 3
64. Zhang, C., Cui, Z., Zhang, Y., Zeng, B., Pollefeys, M., Liu, S.: Holistic 3d scene understanding from a single image with implicit representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8833–8842 (2021) 4
65. Zhang, J., Huang, W., Peng, B., Wu, M., Hu, F., Chen, Z., Zhao, B., Dong, H.: Omni6dpose: A benchmark and model for universal 6d object pose estimation and tracking. In: ECCV (2024) 3
66. Zhang, L., et al.: Clay: A controllable large-scale generative model for creating high-quality 3d assets. TOG (2024) 4
67. Zhang, X., Chen, Z., Wei, F., Tu, Z.: Uni-3d: A universal model for panoptic 3d scene reconstruction. In: ICCV (2023) 4