# INNERSCENE: A 4D HAND-OBJECT INTERACTION DATASET FOR DENSELY PACKED INNER STORAGE SCENES

**Anonymous authors**
Paper under double-blind review




**Figure 1: InnerScene captures both densely arranged storage scenes and the motions that create them.** The first four panels show representative refrigerator, utility-cabinet, pantry-cabinet, and kitchen-cabinet scenes, with textured observations above and reconstructed object arrangements below. The right panel shows the estimated hand trajectory together with the placed-object trajectory; this hand–object motion annotation is a central part of the 4D data, alongside the static scene geometry and synchronized views. For multi-shelf displays, the lower rendering corresponds to the upper shelf.

## ABSTRACT

3D scene data increasingly support perception, generation, and manipulation, yet densely stocked inner-container scenes remain difficult to capture with the geometry and action history these tasks jointly require. A stocked cabinet is the result of many placements, but its final state hides the insertion paths through shrinking free space that produced it; walls and clutter conceal even the objects inside. Directly synthesizing this linked 4D record is difficult because each placement changes the geometry and demands plausible hand–object contact. We introduce InnerScene, to our knowledge the first large-scale, real-captured 4D dataset of densely packed inner-container scenes linking reconstructed object assets and metric layouts to synchronized multi-view observations and estimated hand–object placement trajectories. We record placements progressively in an open capture setup that preserves container spatial constraints, then register objects and motion in a shared metric frame. The resulting data support three practical uses: conditional layout generation populates virtual cabinets with specified objects for retrieval and rearrangement research; single-view instance-level reconstruction turns a cabinet-opening image into a 3D inventory for reasoning about access; and human-to-robot trajectory transfer turns selected recorded insertions into executed robot placements in crowded cabinets. InnerScene thus ties what is stored, where it is stored, and how it was placed into one resource for building, understanding, and acting in inner-container scenes.

## 1 INTRODUCTION

Indoor datasets now provide increasingly rich geometry, object instances, layouts, and human motion, from ScanNet and Matterport3D to tabletop resources such as TO-Scene and MesaTask (Dai et al., 2017; Chang et al., 2017; Yeshwanth et al., 2023; Xu et al., 2022; Hao et al., 2025). Yet

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

densely packed inner-container scenes remain difficult: walls and clutter hide objects, each placement changes free space, and the hand–object motion that created the final arrangement disappears. We introduce InnerScene, a real-captured 4D resource that links reconstructed object assets, metric layouts, synchronized views, and human insertion trajectories in one storage setting.

Cabinets and drawers combine stacking, close contact, restricted openings, and changing free space. Simulation must keep object identities, poses, observations, and hand contact consistent through every placement, so we instead record real placements and annotate their synchronized videos (Figure 2).

Turning a placement video into a coherent 4D annotation is difficult: the same physical object must keep its identity and metric pose as more objects are added to the scene, and its insertion must be paired with the moving hand. **(1) Container occlusion:** opaque walls block the views needed to reconstruct objects deep inside. We therefore express the container as a footprint, access opening, and height limit in an open capture rig, enabling eight synchronized cameras to observe the physical arrangement. **(2) Progressive object occlusion:** even with open sides, objects occlude one another as the container fills. Rather than annotate only the finished scene, we record each placement and observe the newly settled object from synchronized views before later objects cover it. Each physical object is separately reconstructed as a 3D asset before motion tracking; after metric alignment and stable-state registration, the asset supplies the shape and persistent identity needed to recover its insertion trajectory. **(3) Metric inconsistency:** depth inferred separately for different views and times can disagree in scale. We keep camera poses fixed and align each depth output to a board-anchored metric frame through same-image depth agreement and geometric reprojection across views and already settled objects over time. **(4) Dynamic tracking:** the hand and moving object may vanish from individual views, and tracking errors can yield plausible but incorrect 3D trajectories. We track the object independently in each view, select the best visible estimate, and register hand motion into the same world frame. An automated visual reviewer screens each complete hand–object trajectory using overlays of the reprojected hand skeleton and object mesh.

InnerScene contains 10,821 fully arranged storage scenes, each counted after all objects in one placement sequence have been added, and 23,814 reconstructed object models in one-to-one correspondence with physical objects. Approximately 1.9 million images count the eight views at placement-completion moments, rather than every frame of the videos. For each scene, the finished object arrangement is linked to its intermediate arrangements and the retained joint hand–object trajectories of the placements that formed it, all in a shared metric coordinate frame.

These linked observations support three ways to build, perceive, and act within densely packed storage scenes. *Conditional layout generation* takes a container and specified object assets and fills it with a plausible arrangement, creating simulated cabinets that could support retrieval and rearrangement research. *Single-view instance-level reconstruction* turns a cabinet-opening image into separate 3D objects and their arrangement, providing a candidate digital inventory for reasoning about access to stored items. *Human-to-robot trajectory transfer* turns a recorded insertion through clutter into a robot placement path, giving the temporal annotation a direct route to physical cabinet restocking.

Our contributions are: (1) a large dataset linking fully arranged storage scenes, reconstructed instance assets, synchronized observations, and placement trajectories; (2) a capture and annotation pipeline that combines progressive observation, metric depth alignment, and complementary-view tracking, with explicit quality screening; and (3) experiments on layout generation, instance-level reconstruction, and robot placement transfer that demonstrate the uses and current limits of this supervision.

## 2 Related Work

**Indoor, tabletop, and storage datasets.** Indoor resources span reconstructed rooms, furnished scenes, and tabletop arrangements (Dai et al., 2017; Chang et al., 2017; Yeshwanth et al., 2023; Fu et al., 2021; Xu et al., 2022; Hao et al., 2025). Inner-container scenes have also been studied: ESKO6d records stored kitchen objects and includes progressive placement sequences and object models (Richter-Klug et al., 2019); FridgeNet studies occlusion using synthetic refrigerator images (Koporec & Perš, 2019); and KITchen provides real kitchen observations with object-pose annotations (Younes & Asfour, 2024). MessyKitchens addresses contact-rich object-level recon-

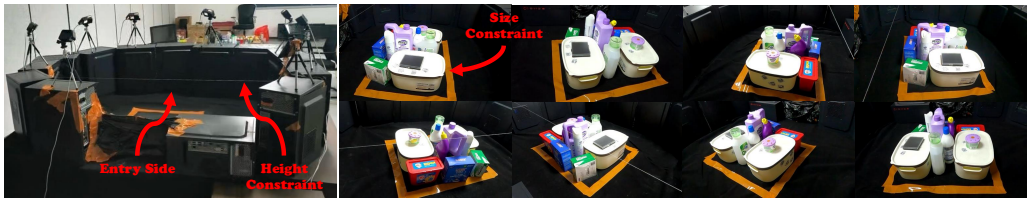



Figure 2: **Capture setup and synchronized observations.** The capture rig represents a storage region with an explicit footprint/size constraint, designated entry side, and height constraint while leaving the contents visible to eight surrounding cameras. The right panel shows synchronized views of progressive object placement. Objects and human placements are real; the enclosure constraints are represented by the capture rig rather than a fully enclosed cabinet.

struction (Ansari et al., 2026). These works establish the setting; InnerScene adds densely arranged scenes linked to multi-view placement histories. Appendix B compares acquisition units explicitly.

**Scene generation and reconstruction.** Layout generators such as ATISS, DiffuScene, CabiNet, and InternScenes model indoor or cabinet arrangements (Paschalidou et al., 2021; Tang et al., 2024; Murali et al., 2023; Zhong et al., 2026). Image-conditioned methods include TRELLIS, SAM 3D, and PartCrafter (Xiang et al., 2024; Chen et al., 2025; Lin et al., 2025b); we use them to test reconstruction under storage occlusion. Our contribution is captured supervision linking arrangements to placement processes.

**Hand–object data and human demonstrations for robots.** HOI4D, ARCTIC, TACO, HOT3D, and GigaHands provide complementary resources for hand–object interaction and motion (Liu et al., 2022; Fan et al., 2023; Liu et al., 2024; Banerjee et al., 2025; Fu et al., 2025). EgoDex studies dexterous manipulation using large-scale egocentric video (Hoque et al., 2026). EgoMimic combines human and robot demonstrations for imitation learning (Kareer et al., 2024), and MimicPlay uses human play to guide robot behavior (Wang et al., 2023). Human2Sim2Robot extracts motion from a human demonstration and uses simulation-based reinforcement learning for transfer (Lum et al., 2025). InnerScene contributes synchronized external views of how densely arranged storage scenes form through successive placements, with hand motion and object assets registered to a shared scene. Our robot experiment uses extracted trajectories directly after retargeting; it does not train or evaluate a policy.

## 3 The InnerScene Dataset

We convert each progressive placement sequence into metric object states, object trajectories, hand poses, and hand–object relationships. Building on the first-version capture and registration pipeline, we add sequence-level view selection, depth-anchored hand motion, and hand–object contact refinement for dense occlusion.

### 3.1 Capture setup and progressive acquisition

Densely packed storage is hard to capture with a conventional closed cabinet: the walls hide objects and hands, while each newly placed object blocks the views that were useful for the previous placement. We therefore use an open capture rig whose tabletop footprint, designated opening, and height limit reproduce the dominant insertion and volume constraints while keeping the contents observable. Eight synchronized RGB cameras surround the rig (Figure 2); the redundant views are necessary because no single camera remains reliable throughout a placement.

Objects are placed sequentially through the opening. A finished image alone cannot reveal when an object became stable or how it entered the container, so we retain both settled frames and the intervening manipulation. The settled frame gives a clean registration target before later objects cause occlusion; the motion frames provide the hand and object evidence needed for a trajectory. Each physical object also receives a smartphone orbit video for a textured instance asset, while a calibration board fixes camera geometry and metric scale (Appendix C).

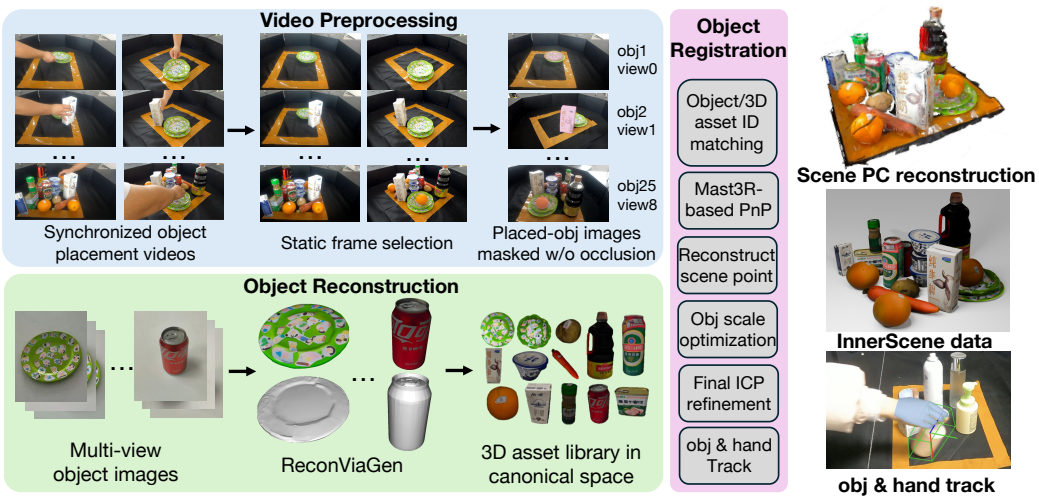



Figure 3: **From recorded placements to associated 4D annotations.** Synchronized placement videos are preprocessed to select stable states and placed-object observations. In parallel, multi-view object images are reconstructed into a canonical asset library. Asset identity matching, MASt3R-based PnP, scene point reconstruction, object-scale optimization, and final ICP register each asset in the scene frame; the resulting object and hand tracks provide the joint 4D annotation.

### 3.2 A LINKED REPRESENTATION OF SCENES AND MOTION

For each scene, synchronized observations, textured assets, persistent identities, masks, and object transformations share a world frame $W$. At stable times, the arrangement is

$$\mathcal{S}_k = \{(i, M_i, T_i^W(\tau_k)) : i \text{ has been placed by } \tau_k\}, \tag{1}$$

where $T_i^W(t) \in \mathrm{SE}(3)$ maps the metric asset into the scene. During placement $i$, the annotation links $T_i^W(t)$, hand joints $H_i^W(t)$, and their synchronized observations, producing one joint trajectory in the shared frame (Figure 3).

### 3.3 SPATIOTEMPORAL METRIC GEOMETRY AND INSTANCE REGISTRATION

**Complementary geometry estimates.** Depth predictions from different cameras and timestamps are not automatically metrically consistent, which is especially damaging when an object is compared across successive placements. We therefore run VGGT (Wang et al., 2025a) on synchronized stable views and Depth Anything 3 (Lin et al., 2025a) on each camera sequence, then use the calibration board to anchor both sources. The remaining per-output scale differences are solved jointly below so that the same object has one metric interpretation across views and time.

**Scale-only consistency optimization.** Let $q$ index an individual depth output from a model, camera, and prediction time, and let $D_q(u)$ denote its depth at pixel $u$. Each such output has a scalar $s_q$, giving $\widehat{D}_q(u) = s_q D_q(u)$. The calibrated camera-to-world transformation $(R_c, t_c)$ and intrinsics $K_c$ remain fixed. A back-projected point is

$$X_q(u; s_q) = R_{c(q)}\big(s_q D_q(u) K_{c(q)}^{-1} \bar{u}\big) + t_{c(q)}, \tag{2}$$

with homogeneous pixel coordinate $\bar{u}$. We fit the scales using same-image depth agreement and geometric correspondences across views or settled times:

$$\begin{aligned} \min_{\boldsymbol{s} \in \mathcal{S}_{\mathrm{cal}}} \quad & \lambda_d \sum_{(q,r,u) \in \mathcal{P}} \big(s_q D_q(u) - s_r D_r(u)\big)^2 \\ & + \lambda_g \sum_{(q,r,u,v) \in \mathcal{C}} \|X_q(u; s_q) - X_r(v; s_r)\|_2^2 . \end{aligned} \tag{3}$$

Here $\mathcal{P}$ and $\mathcal{C}$ are valid same-image and geometric correspondence sets, and $\mathcal{S}_{\mathrm{cal}}$ is anchored by the calibration board. Camera poses and additive depth offsets remain fixed; implementation details are given in Appendix C.3.

**Asset association and registration.** A reconstructed mesh is useful only if it refers to the same physical instance across stable states; appearance-only retrieval can confuse similar containers and registration can absorb an incorrect asset identity into the pose. We therefore segment orbit images with Grounding DINO and SAM 2 (Liu et al., 2023; Ravi et al., 2024), reconstruct textured assets with ReconViaGen (Chang et al., 2025), and match scene instances using DINO features (Caron et al., 2021). The retrieved identity is kept fixed across all later states, so temporal motion is estimated for one rigid asset rather than for a newly reconstructed shape at every frame.

Registration then separates initialization from metric refinement. MASt3R/PnP provides a pose hypothesis from rendered asset views and the observed instance; the depth-scale solution places the hypothesis in metric units, and point-to-point ICP removes the remaining surface misalignment. We reject candidates with insufficient visible support before ICP, because optimizing a largely occluded or incorrectly matched mesh can produce a low residual for the wrong pose. The surviving transform is stored in $W$ and reused as the geometric reference for object tracking and hand–object contact refinement (Appendix C.2).

### 3.4 Temporal hand–object annotation

**Object motion from complementary views.** Object tracking is difficult because a short occlusion can turn a plausible pose into an irrecoverably wrong trajectory. We therefore do not treat a single tracker output as ground truth. FoundationPose (Wen et al., 2024) runs in two modes: at the first visible frame it generates and refines pose hypotheses from the registered mesh, mask, RGB image, and aligned depth; in later frames the previous pose initializes temporal tracking and local refinement. This preserves temporal continuity while allowing independent camera views to provide recovery when one view becomes ambiguous.

For near-axisymmetric objects, we lock spin to the keyframe estimate while leaving the symmetry-axis direction free, removing only the unobservable motion. Each pose is mapped to $W$ with the fixed camera transform.

A second failure mode is view switching: selecting a different camera at each frame creates discontinuous motion even when each individual estimate is plausible. For each placement sequence, we discard the four views with the smallest target-mask areas, score the remaining views by FoundationPose confidence logits, and select one winner for the complete sequence. This sequence-level decision prevents frame-wise switching and yields one temporally coherent exported trajectory. The raw target-mask pixel count

$$A_c(t) = \sum_u \mathbf{1}[u \in \mathcal{M}_{i,c}(t)] \tag{4}$$

provides the visibility heuristic. If $c^*$ is the selected sequence-level view, the exported object pose is

$$T_i^W(t) = T_{c^*}^W \, T_i^{c^*}(t). \tag{5}$$

The selected estimate is exported in $W$; mask area is used as a direct visibility heuristic rather than a surface-visibility fraction.

**Hand motion in the shared frame.** Hand tracking has a different failure mode: Dyn-HaMR (Yu et al., 2025) provides coherent RGB motion, but its world-space scale can drift without metric depth. We use its SLAM component for temporal motion and anchor the SLAM keypoints to calibrated depth and fixed camera geometry. The anchor supplies metric scale and camera placement while preserving temporal motion, so hand and object poses share one frame and timestamps.

**Hand–object contact refinement.** Independent hand and object estimates can each reproject well while disagreeing at the grasp, which is the failure mode most relevant to trajectory transfer. After view selection, the object pose is fixed and only the hand is optimized for reprojection and surface contact, following MagicHOI (Wang et al., 2025b). Fixing the object prevents contact correction from moving the registered scene geometry; optimizing the hand resolves local grasp and release inconsistencies while retaining the measured object trajectory.

**Automatic joint-trajectory screening.** For quality review, we reproject the hand skeleton and object model onto the captured images and submit the overlays to a VLM. A failed review removes the complete joint hand–object trajectory for that placement. Across the processed data, 80.31% of

Table 1: **Comparison with representative storage and object-arrangement resources.** Abbreviations: 'Type' is the acquisition setting; 'Sc.'/'Seq.'/'Img.' are scenes/sequences/images; 'Obj.'/'Cls.' are objects/categories; 'BB' is 3D bounding-box supervision; 'M' is an object model; 'H/O' is hand–object motion. Counts retain each resource's native unit and are not directly interchangeable. 'MV' denotes multi-view observations, and '–' denotes unreported or out-of-scope supervision.

| Data | Type | Sc. | Img./Seq. | Obj./Cls. | BB | M | MV | H/O |
|---|---|---|---|---|---|---|---|---|
| KITchen (Younes & Asfour, 2024) | Real | 2 | 205K/– | 111/– | ✓ | – | MV | – |
| ESKO6d (Richter-Klug et al., 2019) | Real | 12 | –/47 | –/– | ✓ | ✓ | MV | Seq. |
| FridgeNet (Koporec & Perš, 2019) | Synth. | 36 | 95K/– | –/36 | – | – | – | – |
| TO-Scene (Xu et al., 2022) | Tabletop | 20.7K | –/– | –/– | ✓ | ✓ | – | – |
| MesaTask (Hao et al., 2025) | Tabletop | 10.7K | –/– | –/6 | ✓ | ✓ | – | Episodes |
| InnerScene | Real | 10,821 | 1.9M/– | 23,814/8 | ✓ | ✓ | 8 | H/O |

joint trajectories are retained and 19.69% are rejected. The denominator counts placement trajectories once, independently of the number of views. This rate measures the outcome of the screening procedure; it is not an independent measurement of 3D tracking accuracy. Review configuration is given in Appendix C.4.

### 3.5 Dataset scale, splits, and annotation evidence

Table 1 compares representative resources. InnerScene contains 10,821 fully arranged scenes, 23,814 reconstructed object assets, eight synchronized views per selected state, and approximately 1.9 million stable-state images. All views, intermediate states, and trajectories from one scene share a split; 80 scenes form the layout test set and the remainder use a 9:1 train/validation split, with no identical reconstructed test mesh shared with training.

**Release scope.** We plan to release the storage scenes, object assets, synchronized videos, retained joint trajectories, split definitions, and processing code.

## 4 Applications

The applications test whether the annotations remain useful after capture: can they produce physically plausible new storage layouts, recover an object-level inventory from the view available to a robot, and transfer a human insertion into a real cabinet? Each task stresses a different link in the dataset—static arrangement, occluded geometry, or time-indexed hand–object motion.

### 4.1 Conditional layout generation

A final cabinet state is insufficient for planning retrieval: a useful simulator must place known objects in reachable, supported, and collision-free configurations before any new action is executed. We therefore condition layout generation on container dimensions and reconstructed instance assets, first predicting support relations and coarse poses, then refining each pose with object geometry and applying rigid-body relaxation to remove residual collisions (Appendix E). This uses the dataset as metric supervision for both what is stored and how objects can support one another.

The test asks whether these constraints survive outside the training scenes. On 80 held-out InnerScene scenes, we compare against Rule-Stack, extreme-point packing, and direct-layout GPT-6 and Gemini-3.5 using the same object sets and containers. Table 2 shows a consistent advantage for our pipeline: it ranks first (1.80), wins 56.7% of comparisons, and reaches the best score on every visual criterion while maintaining 100% reported physical stability. The margin over Rule-Stack is largest in density and support, indicating that the gain comes from modeling relations between object geometry and support structure rather than merely producing plausible-looking stacks.

We then remove dataset-specific object identities: the Objaverse test uses 100 externally designed textured objects, 50 from unseen categories, and ten unseen box-like containers. Each container holds 20–35 objects. In the bottom two rows of Figure 4, our layouts retain support, compactness, and stacking while competing methods exhibit floating, tilting, or unused space. The result shows that the captured supervision transfers through geometry and support relations rather than memorizing the InnerScene asset catalog; the complete protocol is in Appendix F.

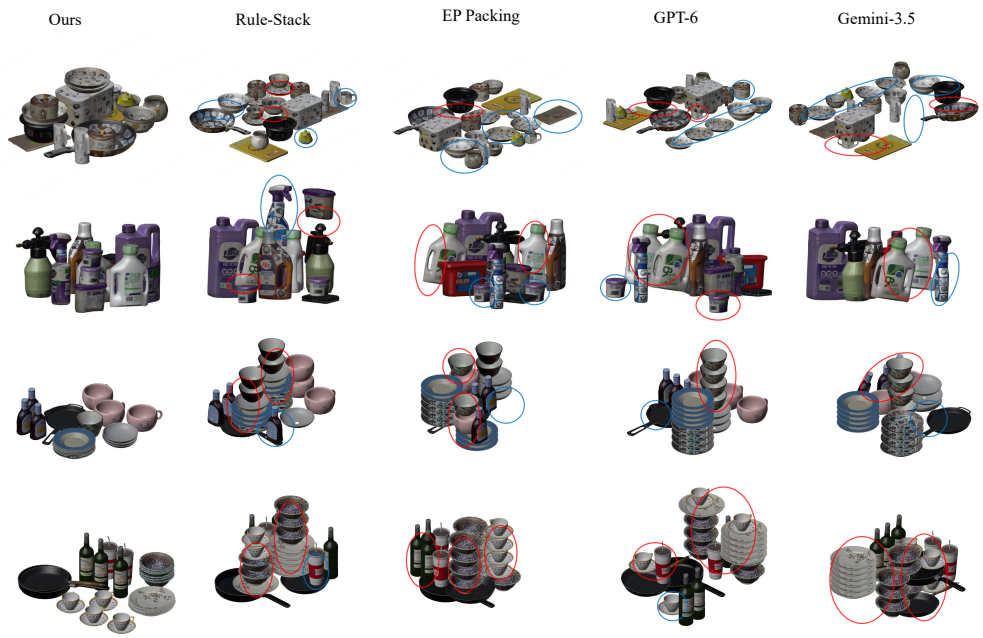



Figure 4: **Task 1: conditional layout generation.** Each column is a method: Ours, Rule-Stack, EP Packing, GPT-6, and Gemini-3.5. The top two rows show held-out InnerScene test scenes. The bottom two rows show the separate Objaverse generalization test with externally designed objects and unseen box-like containers. For the forth row, cup and saucer are treated as a single mesh. Red circles mark representative physical artifacts, including floating objects, unstable poses, incorrect support relations, and severe tilting; blue circles mark semantic or layout artifacts, such as weak grouping, overly flat arrangements, insufficient stacking, and unused gaps.

Table 2: **Task 1 layout generation on the held-out InnerScene test set.** R: realism; D: density; S: support; P: placement; Sem: semantic plausibility. The five judgment scores use the reported 1–5 scale; Physical is the reported object-stability ratio. The qualitative comparison in Figure 4 additionally includes the separate Objaverse generalization setting in its bottom two rows.

| Method | R | D | S | P | Sem | Rank ↓ | First ↑ | Physical ↑ |
|---|---|---|---|---|---|---|---|---|
| Ours | **4.10** | **4.33** | **3.83** | **4.03** | **4.07** | **1.80** | **56.7%** | **100%** |
| Rule-Stack | 3.20 | 3.77 | 2.90 | 3.17 | 3.17 | 2.53 | 23.3% | 66.3% |
| EP Packing | 2.60 | 2.47 | 2.83 | 2.70 | 2.60 | 3.23 | 3.3% | **100%** |
| GPT-6 | 2.27 | 2.10 | 2.97 | 2.67 | 2.27 | 3.60 | 10.0% | 61.2% |
| Gemini-3.5 | 2.07 | 2.03 | 2.47 | 2.27 | 2.03 | 3.83 | 6.7% | 56.3% |

### 4.2 Single-view instance-level 3D reconstruction

A robot often sees a cabinet only through its opening, so deciding whether an object is reachable requires more than image-level recognition: it needs separate shapes and their spatial arrangement under severe occlusion. We therefore reconstruct an instance-level inventory from one opening-side image. The protocol supplies the object count and compares TRELLIS.2 (Xiang et al., 2024), SAM 3D (Chen et al., 2025), and PartCrafter (Lin et al., 2025b); our adaptations are documented in Appendix G.

Table 3 shows that dataset supervision improves both assembly and object quality: fine-tuned PartCrafter reduces scene CD from 0.286 to 0.123 and raises object F-score from 0.453 to 0.862, the best results in both object columns. Tuned TRELLIS.2 obtains the best scene F-score (0.843), while SAM 3D is competitive on object geometry but weaker on scene assembly. Thus the linked assets and layouts provide useful supervision even when only one occluded view is available; remaining failures are missing or hallucinated instances and inaccurate spatial alignment.

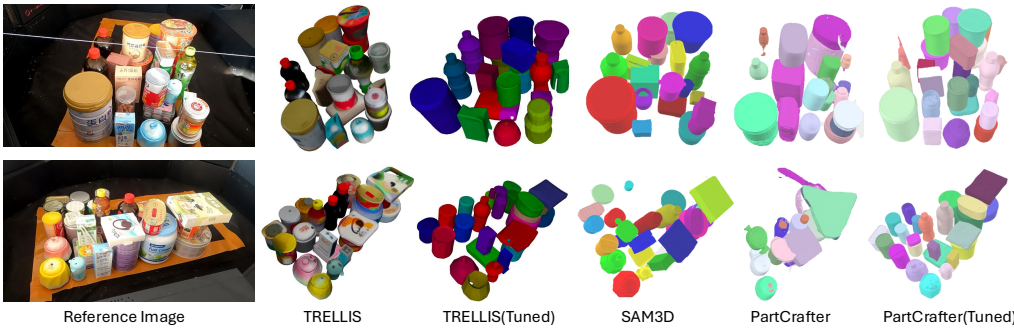



Figure 5: **Task 2: single-view instance-level 3D reconstruction.** Each example starts from a reference image and compares TRELLIS.2, instance-color adapted TRELLIS.2, SAM3D, PartCrafter, and PartCrafter fine-tuned on InnerScene. The task evaluates whether a single opening-side view can be converted into separate object geometry and a scene arrangement; the adaptation and fine-tuning settings are described in Appendix G.

Table 3: **Task 2 single-view instance-level reconstruction.** CD is lower-is-better and F-score is higher-is-better. Values retain the original evaluation convention; CD normalization, F-score thresholds, and object matching are specified in Appendix G. A dash indicates that instance-separated object scores were not reported for the original TRELLIS.2 output.

| Method | Scene CD ↓ | Scene F ↑ | Object CD ↓ | Object F ↑ |
|---|---|---|---|---|
| TRELLIS.2 | 0.132 | 0.833 | – | – |
| TRELLIS.2 (Tuned) | <u>0.128</u> | **0.843** | 0.132 | 0.543 |
| SAM3D | 0.174 | 0.632 | <u>0.121</u> | <u>0.784</u> |
| PartCrafter | 0.286 | 0.373 | 0.205 | 0.453 |
| PartCrafter (Tuned) | **0.123** | <u>0.832</u> | **0.118** | **0.862** |

### 4.3 Human-to-robot placement trajectory transfer

A layout is useful for manipulation only if the recorded action that created it can be replayed through the same constrained opening. We therefore retarget the synchronized hand–object trajectories, together with the registered scene geometry, to a GALAXEA R1 Lite right-arm gripper. An operator loads each object before the placement segment, and the supplementary video shows executions in cleaning-supply and kitchenware cabinets. The demonstrations preserve the insertion order and contact-driven path through clutter, providing qualitative evidence that the dataset connects perception to action; autonomous pickup, policy learning, and comparative success rates are outside this application.

## 5 Annotation Pipeline Ablation

We evaluate the annotation pipeline at two levels. Table 4 reports object pose and geometry on controlled rendered inputs with known reference geometry, and evaluates the hand branch on a separately rendered dynamic benchmark. Following the synthetic hand–object construction used by ForeHOI (Chen et al., 2026), we use textured Objaverse object meshes (Deitke et al., 2022), GraspXL-generated hand grasp motions (Zhang et al., 2024), and textured MANO hands to render multiview RGB–D videos. Rendering provides exact per-frame ground-truth poses for both the hand and object, so every ablation is compared against the same known motion rather than against an estimated label. The controlled validation measures annotation error under this synthetic protocol; it does not certify the accuracy of the full real-capture corpus or dynamic hand–object trajectories.

For the dynamic benchmark, hand pose accuracy is measured by MPJPE, the mean Euclidean distance between corresponding estimated and ground-truth hand joints in millimeters. We report the relative-scale error (RS) used by MagicHOI (Wang et al., 2025b): if $s$ is the scale ratio obtained by ICP alignment of the reconstructed object to its ground-truth mesh, $RS = 1/s - 1$ for $s < 1$ and $RS = 1 - 1/s$ otherwise. Thus $RS = 0$ denotes the correct physical scale and larger values indicate stronger scale mismatch. SIV is the solid intersection volume between the hand and object surfaces, measured in $cm^3$; it penalizes hand–object penetration and is the volume-based counterpart

Table 4: **Annotation pipeline ablation.** (a) Object pose and geometry. (b) Temporal hand–object annotation.

**(a) Object pose and geometry**

| Variant | Trans. (cm) ↓ | Rot. (°) ↓ | CD (cm) ↓ |
|---|---|---|---|
| w/o depth-scale alignment | 1.32 | 10.54 | 1.64 |
| w/o asset registration | 0.98 | 6.12 | 2.34 |
| w/o final ICP | 0.54 | 1.24 | 1.53 |
| Full static pipeline | **0.18** | **0.89** | **0.23** |

*Trans.* and *Rot.* are object translation and rotation errors; CD is absolute-scale geometry Chamfer distance. The validation uses four artist-constructed scenes and 50 objects rendered as pseudo-input videos.

**(b) Temporal hand–object annotation**

| Variant | Hand MPJPE (mm) ↓ | RS ↓ | SIV ($cm^3$) ↓ |
|---|---|---|---|
| Monocular RGB | 25.93 | 0.78 | 20.43 |
| + view selection | 18.43 | 0.83 | 18.43 |
| + depth anchor | **5.32** | 0.19 | 10.92 |
| + contact opt. | 6.04 | **0.11** | **7.34** |

All metrics are lower-is-better; definitions are given in the Ablation text.

Table 5: **Real-capture trajectory coverage and collection effort.** Retention counts complete hand–object trajectories once after overlay review. Placement times cover only the five-scene, 25-objects-per-scene comparison; they exclude calibration, object capture, reconstruction, registration, computation, and review.

| Evidence | Value | Scope |
|---|---|---|
| Retained joint trajectories | 80.31% | complete trajectories after overlay review |
| Rejected joint trajectories | 19.69% | complete trajectories after overlay review |
| Physical placement | 30 s/scene | five scenes, 25 objects/scene |
| Blender placement | 5 min/scene | experienced-artist comparison |
| Complete annotation cost | not measured | includes capture, processing, and review |

to contact consistency, following CPF (Yang et al., 2021). All three metrics are lower-is-better and are computed against the rendered ground truth.

The current static result is a controlled rendered validation, not a corpus-wide accuracy claim. The complementary real-capture evidence is summarized in Table 5. Its retention rate counts complete hand–object trajectories after overlay review: 80.31% are retained and 19.69% are rejected, with each placement counted once regardless of the number of camera views. The effort comparison is narrower: 30 s/scene for physical placement and 5 min/scene for Blender placement measure only the placement stage for the stated five-scene comparison. They exclude calibration, object capture, reconstruction, registration, computation, and review, so they should not be interpreted as full annotation cost or as a geometric accuracy metric.

## 6 Limitations and Conclusion

InnerScene focuses on rigid objects and regular storage volumes. The capture proxy omits some real-wall interactions, and estimated surfaces or trajectories can remain inaccurate despite consistent reprojections. Mask-area selection is perspective-sensitive and can fail at view switches. Controlled static validation and automatic screening do not establish real trajectory accuracy; robot results cover selected, manually initialized placements.

InnerScene links densely arranged storage scenes to their formation through synchronized observations, metric assets, and hand–object motion. Three applications demonstrate how these associated annotations support layout generation, reconstruction, and placement execution, providing a shared resource for studying storage geometry and interaction.

### AI USE STATEMENT

Generative AI tools assisted manuscript drafting and literature organization. The data-processing pipeline uses learned geometry, reconstruction, and motion-estimation models described in the paper, and a VLM screens joint-trajectory reprojections. The layout application uses a VLM for coarse planning and teacher-generated reasoning traces for fine-tuning; VLM judges are used in its evaluation. Reference entries are inherited from the earlier manuscript or copied from identified source pages, with a separate provenance record and consistency checks.

### ETHICS STATEMENT

The dataset contains recorded human placements and the layout evaluation includes a participant study. The release plan includes synchronized videos and hand-motion annotations, so documentation of participant consent and the treatment of identifying content is part of the release preparation.

### REPRODUCIBILITY STATEMENT

Section 3 defines the capture protocol, counting units, fixed-camera depth alignment, view selection, and trajectory-screening scope. Appendix C–D provides additional data-construction details; Appendix E–H records application settings and inherited experimental results. Author-completion markers identify missing numerical or procedural specifications rather than filling them with assumptions. We plan to release data, retained trajectories, split files, and processing and evaluation code. The public artifact location and complete configuration snapshots will be added when available.

## REFERENCES

Junaid Ahmed Ansari, Ran Ding, Fabio Pizzati, and Ivan Laptev. Messykitchens: Contact-rich object-level 3d scene reconstruction, 2026. URL https://arxiv.org/abs/2603.16868.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Shangchen Han, Fan Zhang, Linguang Zhang, Jade Fountain, Edward Miller, Selen Basol, Richard Newcombe, Robert Wang, Jakob Julian Engel, and Tomas Hodan. HOT3D: Hand and object tracking in 3D from egocentric multi-view videos. *CVPR*, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017.

Jiahao Chang, Chongjie Ye, Yushuang Wu, Yuantao Chen, Yidan Zhang, Zhongjin Luo, Chenghong Li, Yihao Zhi, and Xiaoguang Han. Reconviagen: Towards accurate multi-view 3d object reconstruction via generation. *arXiv preprint arXiv:2510.23306*, 2025.

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.

Yuantao Chen, Jiahao Chang, Chongjie Ye, Chaoran Zhang, Zhaojie Fang, Chenghong Li, and Xiaoguang Han. Forehoi: Feed-forward 3d object reconstruction from daily hand-object interaction videos, 2026. URL https://arxiv.org/abs/2602.06226.

Teodor Gabriel Crainic, Guido Perboli, and Roberto Tadei. Extreme point-based heuristics for three-dimensional bin packing. *INFORMS J. Comput.*, 20:368–384, 2008. URL https://api.semanticscholar.org/CorpusID:2742062.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5828–5839, 2017.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects, 2022. URL https://arxiv.org/abs/2212.08051.

Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24 (6):381–395, 1981.

Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10933–10942, 2021.

Rao Fu, Dingxi Zhang, Alex Jiang, Wanjia Fu, Austin Funk, Daniel Ritchie, and Srinath Sridhar. Gigahands: A massive annotated dataset of bimanual hand activities. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17461–17474, 2025.

Jinkun Hao, Naifu Liang, Zhen Luo, Xudong Xu, Weipeng Zhong, Ran Yi, Yichen Jin, Zhaoyang Lyu, Feng Zheng, Lizhuang Ma, and Jiangmiao Pang. Mesatask: Towards task-driven tabletop scene generation via 3d spatial reasoning, 2025.

Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. Egodex: Learning dexterous manipulation from large-scale egocentric video, 2026. URL https://arxiv.org/abs/2505.11709.

Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video, 2024. URL https://arxiv.org/abs/2410.24221.

Gregor Koporec and Janez Perš. Deep learning performance in the presence of significant occlusions - an intelligent household refrigerator case. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 2532–2540, 2019. doi: 10.1109/ICCVW.2019.00310.

Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Ep n p: An accurate o (n) solution to the p n p problem. *International journal of computer vision*, 81(2):155–166, 2009.

Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with mast3r, 2024.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025a.

Yuchen Lin, Chenguo Lin, Panwang Pan, Honglei Yan, Yiqiang Feng, Yadong Mu, and Katerina Fragkiadaki. Partcrafter: Structured 3d mesh generation via compositional latent diffusion transformers. *arXiv preprint arXiv:2506.05573*, 2025b.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

Yun Liu, Haolin Yang, Xu Si, Ling Liu, Zipeng Li, Yuxiang Zhang, Yebin Liu, and Li Yi. Taco: Benchmarking generalizable bimanual tool-action-object understanding. *arXiv preprint arXiv:2401.08399*, 2024.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21013–21022, 2022.

Tyler Ga Wei Lum, Olivia Y. Lee, C. Karen Liu, and Jeannette Bohg. Crossing the human-robot embodiment gap with sim-to-real rl using one human demonstration, 2025. URL https://arxiv.org/abs/2504.12609.

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers, 2024. URL https://arxiv.org/abs/2401.08740.

Adithyavairavan Murali, Arsalan Mousavian, Clemens Eppner, Adam Fishman, and Dieter Fox. CabiNet: Scaling neural collision detection for object rearrangement with procedural scene generation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2023. URL https://arxiv.org/abs/2304.09302.

Despoina Paschalidou, Amlan Kar, Maria Shugrina, Karsten Kreis, Andreas Geiger, and Sanja Fidler. Atiss: Autoregressive transformers for indoor scene synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL https://arxiv.org/abs/2408.00714.

Jesse Richter-Klug, Constantin Wellhausen, and Udo Frese. Esko6d - a binocular and rgb-d dataset of stored kitchen objects with 6d poses. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 893–899, 2019. doi: 10.1109/IROS40897.2019.8967937.

Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Denoising diffusion models for generative indoor scene synthesis, 2024. URL https://arxiv.org/abs/2303.14207.

Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a.

Shibo Wang, Haonan He, Maria Parelli, Christoph Gebhardt, Zicong Fan, and Jie Song. Magichoi: Leveraging 3d priors for accurate hand-object reconstruction from short monocular video clips, 2025b. URL https://arxiv.org/abs/2508.05506.

Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17868–17879, 2024.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024.

Mutian Xu, Pei Chen, Haolin Liu, and Xiaoguang Han. To-scene: A large-scale dataset for understanding 3d tabletop scenes. In *ECCV*, 2022.

Lixin Yang, Xinyu Zhan, Kailin Li, Wenqiang Xu, Jiefeng Li, and Cewu Lu. Cpf: Learning a contact potential field to model the hand-object interaction, 2021. URL https://arxiv.org/abs/2012.00924.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12–22, 2023.

Abdelrahman Younes and Tamim Asfour. Kitchen: A real-world benchmark and dataset for 6d object pose estimation in kitchen environments. In *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pp. 803–810. IEEE, 2024.

Zhengdi Yu, Stefanos Zafeiriou, and Tolga Birdal. Dyn-hamr: Recovering 4d interacting hand motion from a dynamic camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

Hui Zhang, Sammy Christen, Zicong Fan, Otmar Hilliges, and Jie Song. Graspxl: Generating grasping motions for diverse objects at scale, 2024. URL https://arxiv.org/abs/2403.19649.

Yujia Zhang, Xiaoyang Wu, Yunhan Yang, Xianzhe Fan, Han Li, Yuechen Zhang, Zehao Huang, Naiyan Wang, and Hengshuang Zhao. Utonia: Toward one encoder for all point clouds. In *ICML*, 2026.

Weipeng Zhong, Peizhou Cao, Yichen Jin, Li Luo, Wenzhe Cai, Jingli Lin, Hanqing Wang, Zhaoyang Lyu, Tai Wang, Bo Dai, Xudong Xu, and Jiangmiao Pang. Internscenes: A large-scale simulatable indoor scene dataset with realistic layouts, 2026. URL https://arxiv.org/abs/2509.10813.

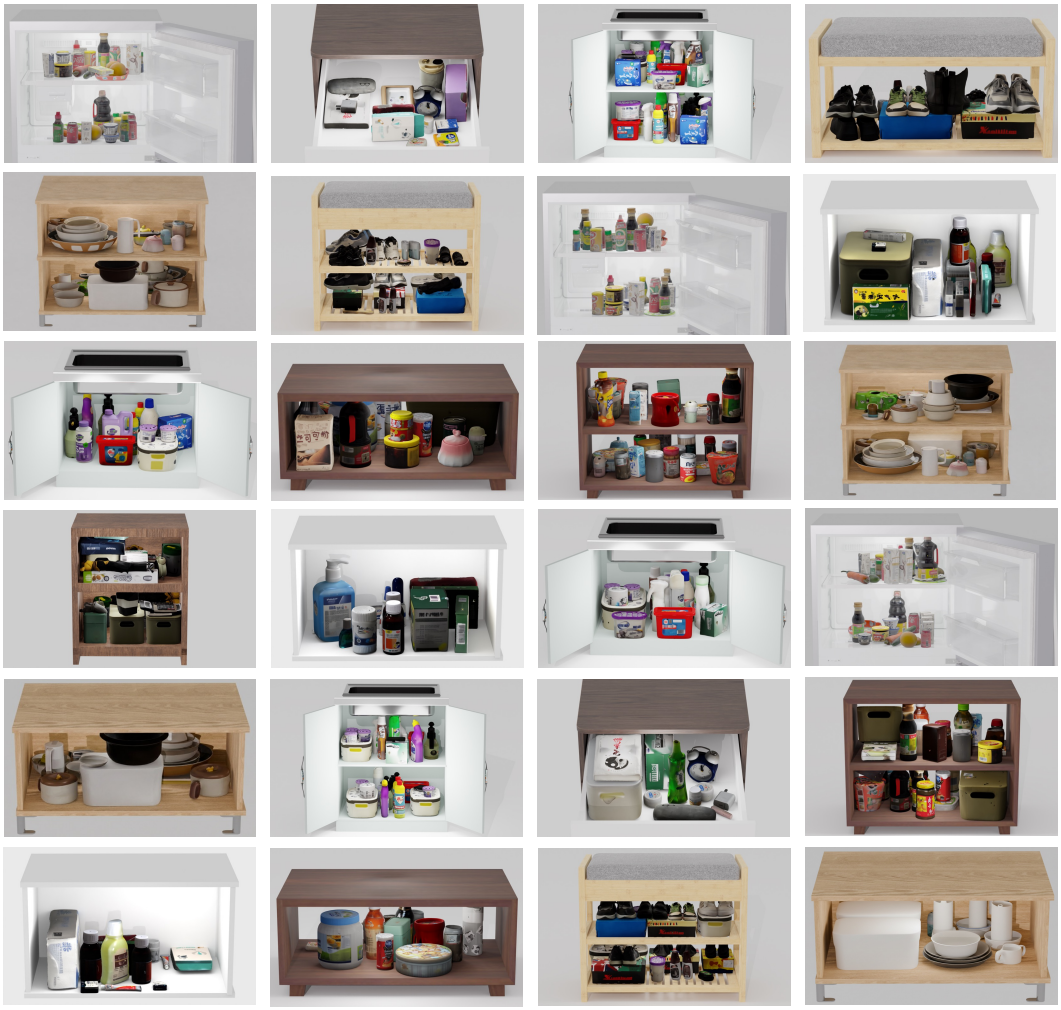


Figure 6: **Dataset gallery.** Representative densely packed storage scenes spanning refrigerators, cabinets, and open shelving. The gallery shows the diversity of containers and object arrangements represented in InnerScene.

# A Dataset Gallery

# B Scope of Related Data Resources

Table 6 compares the scope of representative resources without equating their acquisition units. In particular, ESKO6d contains 12 scenes and 47 sequences for each sensor modality; its sequence count should not be reported as a scene count. FridgeNet's 36 object categories are also distinct from its approximately 95,000 synthetic images. KITchen includes approximately 205,000 images of 111 objects in two kitchens, with storage settings such as refrigerators, drawers, and dishwashers. Its CAD models are part of the pose-estimation setting. These works should not be described as lacking object models or storage observations.

# C Data Acquisition and Annotation Details

## C.1 Sequence boundaries and stable-state extraction

The recording process includes scene preparation, sequential placement, and removal before the next scene. Briefly covering one camera during scene reset creates a dark interval that can serve as a synchronized sequence delimiter. To identify settled states, the preprocessing stage aggregates hand-detection responses across the views into a temporal activity score. Local minima between

Table 6: **Representative dataset and generation scopes.** References are given in Section 2. Descriptions indicate what motivates the comparison; they do not assert that unlisted annotation types are absent. Sizes retain the original unit.

| Resource | Acquisition focus | Relevant scope |
|---|---|---|
| ESKO6d | Real stored objects | Progressive kitchen-object placement; 12 scenes, pose annotations, and object models. |
| FridgeNet | Synthetic refrigerator images | Occlusion study with about 95K images and 36 object categories. |
| KITchen | Real kitchen observations | About 205K images and 111 objects, with CAD-based pose annotations. |
| MessyKitchens | Real contact-rich scenes | Object-level 3D scene reconstruction, including stacking and close contacts. |
| CabiNet | Procedural cabinet clutter | Large-scale synthetic rearrangement scenes and collision-aware manipulation. |
| InternScenes | Simulatable indoor scenes | Indoor layouts include objects placed in or on cabinets. |
| ARCTIC / TACO | Hand–object interaction | Detailed multi-view interaction data, including bimanual or tool-mediated actions. |
| HOT3D / GigaHands | Hand–object motion | Complementary egocentric or multi-camera hand-motion observations. |
| InnerScene | Real progressive storage placement | 10,821 fully arranged storage scenes associated with instance assets, eight-view observations, and hand–object placement trajectories. |

active placements provide candidate completion timestamps. A sequence containing $N$ placements supplies $N$ such synchronized image groups.

Fixed camera poses make consecutive stable images comparable, but exposure and lighting changes make raw image subtraction unreliable. The static annotation pipeline instead uses a lightweight change-mask predictor with DINO features (Caron et al., 2021) and a shallow convolutional decoder to identify the newly inserted object between successive stable states. This procedure supplies the per-view observations used in asset matching and static registration.

### C.2 Individual object reconstruction and instance association

For every physical object, a smartphone video records an orbit around the object on a tabletop. Twenty frames are sampled uniformly. Grounded segmentation initialized with the prompt `the object on the desk` provides a foreground mask, which SAM 2 propagates through the sampled views (Liu et al., 2023; Ravi et al., 2024). ReconViaGen reconstructs a textured asset from the masked multi-view images (Chang et al., 2025). Its output is a reconstructed model, including estimated surfaces where observations are incomplete.

The matching step compares masked images of the scene instance with masked object images associated with the asset library. A fine-tuned DINO-based feature extractor improves instance discrimination (Caron et al., 2021). Retrieval assigns each scene instance an asset identity, which persists through stable states and temporal annotations. Object reconstruction, instance retrieval, and metric registration are separate operations; a visually plausible mesh does not establish the correctness of its scene pose.

For registration, the asset is rendered from 30 viewpoints on a spherical trajectory. These views and an observed instance image supply a VGGT-based initialization (Wang et al., 2025a). We select a nearby rendered viewpoint, match it to the observation using MASt3R (Leroy et al., 2024), and lift rendered pixels with the known rendered depth. Robust PnP uses these correspondences to initialize the object pose (Lepetit et al., 2009; Fischler & Bolles, 1981). The object's arbitrary reconstruction scale is then aligned with the metric in-scene point map using robust correspondence fitting. Candidate registrations are evaluated across available views; failed candidates are discarded, and a successful candidate with sufficient visible support and alignment fitness initializes point-to-point ICP refinement.

An asset's metric scaling is distinct from the per-output depth scalars in Equation 3. Once the asset is placed in metric units, its scene transformation represents rigid motion. Annotators inspect reconstructed static object arrangements and correct erroneous registrations when necessary.

### C.3 Details needed for metric scale fitting

The camera poses are known and fixed. The variables are multiplicative depth scalars for individual camera/model outputs; there is no optimized camera-pose correction or depth offset. Same-image depth consistency couples VGGT and DA3 predictions, and cross-view or cross-time 3D residuals couple geometrically corresponding observations. A board-derived metric reference fixes the global scale. For temporal residuals, only already-settled objects are eligible. Static background alignment is not assumed.

Equation 3 describes the confirmed least-squares residual families. It deliberately leaves $\mathcal{S}_{\mathrm{cal}}$ abstract until the exact board-anchor implementation is supplied. It also conditions the linear least-squares description on fixed correspondences; reprojection correspondence construction or updates must be specified separately. A geometric consistency loss cannot, by itself, establish absolute accuracy without an independent reference.

### C.4 Temporal tracking and review configuration

FoundationPose (Wen et al., 2024) runs independently for each object and camera viewpoint, with aligned depth and the registered mesh. **1) Keyframe registration.** At the first visible frame, we generate multiple pose hypotheses, refine them, and rank them by the registration confidence logit. **2) Tracking.** Subsequent frames use the previous pose for rotational initialization and local refinement, but translation is reinitialized at every frame from the OSTrack 2D box and the corresponding depth. This prevents translation error from accumulating through recursive warm starts. For near-axisymmetric objects, spin about the object symmetry axis is locked to its keyframe value while the axis direction remains free. **3) View selection and export.** We discard the four views with the smallest target masks, compare confidence logits among the remaining four, and select one winner for the complete placement sequence. The selected pose is transformed to world coordinates with the calibrated camera-to-world transform; we export one pose per frame rather than a confidence-weighted average.

We recover hand motion with Dyn-HaMR (Yu et al., 2025). Its SLAM component provides the temporal estimate from RGB; we anchor the SLAM keypoints to calibrated depth and known camera geometry so that the hand is metric in the shared world frame. We then keep the selected object pose fixed and optimize only the hand pose for reprojection consistency and hand–object contact. This asymmetric refinement corrects hand pose drift without changing the independently registered object trajectory.

For screening, the recorded model name is **Qwen3.7 Plus**. Its inputs are captured images overlaid with the hand skeleton and reprojected object model. A failed review discards the full joint trajectory, with an overall retained proportion of 80.31%. The name is an author-supplied implementation identifier; the Qwen3-VL citation for the layout planner is not used as a substitute technical reference for this reviewer model.

### C.5 Scope of quality measurements

The static validation uses rendered observations of four artist-constructed scenes and 50 objects with known reference geometry. Its reported quantities are translation error 0.18 cm, rotation error $0.89^\circ$, and geometry Chamfer distance 0.23 cm at absolute scale. These are separate from the joint-trajectory retention rate. Neither quantity measures the accuracy of every real trajectory, and passing VLM screening does not certify precise contact geometry.

## D Counting Units, Collection Effort, and Data Release

A fully arranged storage scene is counted once after all objects in that placement sequence have been placed. Intermediate arrangements are part of that same scene, not additional independent scenes. An image is one camera view at one selected completion timestamp; the approximately

1.9 million images sum all eight views. A reconstructed asset is one of 23,814 models in one-to-one correspondence with physical objects. A joint trajectory is one placement's paired hand and object motion, counted once across views. These definitions prevent synchronized observations from inflating the number of independent scenes or demonstrations.

The layout split uses 80 held-out scenes and a 9:1 training/validation split of the remaining scenes. Test meshes are disjoint from training meshes. All intermediate states, views, and trajectories associated with the same complete scene remain in its split. The single-view models use their stated training subsets; final split files should record these subsets and the evaluation scenes explicitly. This data organization makes no claim that a robot policy was trained.

**Provisional duration estimate.** The five-scene timing comparison measures roughly 30 seconds of active placement per scene, with 25 objects in each timing example. Extrapolating this rate to 10,821 scenes gives

$$10{,}821 \times 30/3600 = 90.175 \text{ hours}. \tag{6}$$

Multiplying by an author-specified factor of three to include preparation gives 270.525 hours. This is a provisional estimate, not a sum of video durations, a measurement of interaction-only time, or a measure of total person-hours. Eight synchronized streams do not multiply the elapsed collection time by eight. We keep this estimate out of the headline dataset statistics until recording metadata are audited.

**Planned release.** The intended release associates scene IDs, placement IDs, object IDs, calibrated cameras, synchronized video timestamps, stable-state observations, instance assets, masks, object poses, and retained joint hand–object trajectories. The code release is intended to include preprocessing, alignment, annotation, filtering, application training, and evaluation. Data and code are planned for public release; no public download is claimed to be available in this draft.

# E Layout Application: Reference Method

## E.1 Coarse arrangement and support supervision

The planner receives object images, object bounding-box dimensions, and the container dimensions. The coordinate origin is at the bottom-left-back of the container, with the $Y$ axis aligned with gravity's opposite direction. A Qwen3-VL-8B-Thinking model (Bai et al., 2025) predicts a coarse arrangement and a directed support graph. Training uses scene-level reasoning traces from a Qwen3.5-397B-A17B teacher. The traces describe object affordances, vertical layers, and allocation in the horizontal plane, followed by a structured output.

Ground-reference support relations are constructed from annotated layouts using footprint overlap, vertical ordering, ray-contact tests, and contact-gap tests. A generated example is retained only if its parsed centroids match the annotated centroids within 0.5 cm and its support graph matches exactly. Up to 8,200 scene-level examples are used. LoRA rank, alpha, dropout, learning rate, and training epochs are 64, 128, 0.05, $2 \times 10^{-4}$, and 3, respectively. These settings characterize the reference application, not the creation of the raw captured scenes.

## E.2 Geometry-conditioned refinement

The refiner samples 500 surface points per object after transforming its mesh into the coarse container-frame pose. A frozen Utonia encoder supplies geometric features (Zhang et al., 2026). Variable numbers of objects are concatenated across a mini-batch, with scene membership masks. The reported PackingTransformer configuration has hidden dimension 512, eight attention heads, depth six, MLP ratio four, and approximately 137 million parameters. We adapt a SiT-based backbone (Ma et al., 2024) with alternating intra-object reasoning, support-graph message passing, and inter-object reasoning.

For clean target point coordinates $x_1$ and Gaussian noise $x_0$, the training interpolant and velocity objective are

$$x_t = (1-t)x_0 + tx_1, \qquad \mathcal{L} = \mathbb{E}\left[\left\|v_\theta(x_t, t, F, \mathcal{G}) - (x_1 - x_0)\right\|_2^2\right], \tag{7}$$

Table 7: **Original five-method layout comparison.** R: realism; D: density; S: support; P: placement; Sem: semantic plausibility. All five scores use the reported 1–5 scale. Physical is the reported object stability ratio, not a contact or containment certificate.

| Method | R | D | S | P | Sem | Rank | First | Physical |
|---|---|---|---|---|---|---|---|---|
| Gemini-3.5 | 2.07 | 2.03 | 2.47 | 2.27 | 2.03 | 3.83 | 6.7% | 56.3% |
| GPT-6 | 2.27 | 2.10 | 2.97 | 2.67 | 2.27 | 3.60 | 10.0% | 61.2% |
| Extreme-Point | 2.60 | 2.47 | 2.83 | 2.70 | 2.60 | 3.23 | 3.3% | **100%** |
| Rule-Stack | 3.20 | 3.77 | 2.90 | 3.17 | 3.17 | 2.53 | 23.3% | 66.3% |
| Reference pipeline | **4.10** | **4.33** | **3.83** | **4.03** | **4.07** | **1.80** | **56.7%** | **100%** |

where $F$ denotes geometric features and $\mathcal{G}$ the support graph. Coarse training conditions come from SE(3)-perturbed annotated layouts. The reported curriculum increases rotation/translation noise from 0.05/0.03 to 0.20/0.10. Geometric conditions are dropped with probability 0.1 for classifier-free conditioning. The described inference procedure uses 100 Euler sampling steps.

Predicted points may not form a rigid object. For each object, an SVD-based rigid alignment maps its original sampled points to the prediction:

$$(R_i, t_i) = \arg \min_{R \in \mathrm{SO}(3),\, t \in \mathbb{R}^3} \sum_{k=1}^{500} \|R p_{ik} + t - \widehat{p}_{ik}\|_2^2. \tag{8}$$

Applying this transformation to the original mesh preserves rigidity; it does not guarantee collision-free contact. The reported projection residual, normalized by the object bounding-box diagonal, is 0.015 mean, 0.010 median, 0.040 at the 95th percentile, and 0.10 maximum.

### E.3 Physics-based relaxation

The final stage imports the rigid meshes into Blender/Bullet, uses convex decomposition for collision proxies, and lets the objects settle onto a static supporting plane. The reported translational and rotational damping coefficients are both 0.8. This step addresses residual penetrations or floating objects. Settling is an empirical postprocessing operation, not a mathematical guarantee of valid geometry, object retention, or semantic arrangement.

The method assumes the supplied object geometry is sufficiently accurate for collision checking. Irregular enclosures, extreme occupancy, ambiguous support, and nonrigid objects remain outside or near the limits of the evaluated setting. These are separate from the dataset's motivation and should not be read as general capability limits of scene-generation models.

## F Layout Application: Evaluation and Additional Results

### F.1 Comparison protocols

The original comparison includes Extreme-Point packing (Crainic et al., 2008), a stacking heuristic called Rule-Stack, and direct-layout GPT-6 and Gemini-3.5 baselines. They operate on the same container and object set; prompt-based models receive object information and predict the layout. The reference pipeline is evaluated on the held-out split described in Section 3.5. Table 7 retains the original five-method comparison. The earlier comparison including fine-tuned DiffuScene is a separate experiment; its ranks and selection frequencies are not pooled with the five-method candidate set reported here.

The original VLM evaluation describes a Gemini-2.5-Pro judge, common rendered viewpoints, anonymized candidate labels, random ordering, and repeated evaluation. Scores range from one to five for realism, density, visual support, placement, and overall semantic plausibility. Average rank and first-place frequency summarize relative judgments. These assess rendered appearance rather than independent physical measurements.

**Physical evaluation.** The reported Blender test remeshes the objects, uses convex-decomposed collision proxies and friction 0.8, and perturbs a supporting plane by an angle sampled within $[-2^\circ, 2^\circ]$. An object is marked unstable if it rotates by more than $45^\circ$; Physical is the fraction not marked unstable. A subsequent Isaac Sim check of final layouts also reports 100% stability

Table 8: **Additional VLM-judge evaluations.** Rank is lower-is-better. First-place frequency refers to the relevant candidate set.

| | Reference pipeline | | Strongest baseline | |
|---|---|---|---|---|
| Judge | Rank | First | Rank | First |
| Gemini-3.5-Flash | **1.80** | **56.7%** | 2.53 | 23.3% |
| GPT-5.5 | **1.54** | **63.3%** | 2.72 | 20.0% |
| Claude-Opus-4.8 | **1.50** | **60.0%** | 2.68 | 16.7% |

Table 9: **Additional layout generalization.** Columns use the same score names as Table 7. The standard row is a reference result; the remaining rows belong to the new-object/new-container setting.

| Setting | Method | R | D | S | P | Sem | Physical |
|---|---|---|---|---|---|---|---|
| Standard | Reference | 4.10 | 4.33 | 3.83 | 4.03 | 4.07 | 100% |
| Unseen | Rule-Stack | 3.05 | 3.58 | 2.72 | 3.04 | 3.02 | 61.5% |
| Unseen | Gemini | 2.21 | 2.18 | 2.54 | 2.39 | 2.24 | 52.8% |
| Unseen | Reference | **4.02** | **4.27** | **3.91** | **3.99** | **4.21** | **100%** |

for the reference pipeline. Stability in this protocol does not imply zero interpenetration or verified containment. In particular, the method includes physical relaxation, so its stable outputs cannot be attributed solely to the learned refiner.

### F.2 Multiple judges and participant preferences

Table 8 records the additional anonymized, randomized evaluation with three reported VLM judges. The strongest-baseline entry refers to the best baseline in the corresponding evaluation, not necessarily the same method across judges. Model labels below follow the recorded experiment names; exact serving snapshots belong in the released protocol.

The participant study contains 50 valid participants, 10 held-out scenes, and 1,500 choices. The reported preference shares for the reference pipeline are 72.4% for overall quality, 68.8% for physical plausibility, and 74.2% for arrangement plausibility, averaging 71.8%. These are reported choice shares, not asserted pairwise success rates. Repeated choices from the same participants and scenes are not 1,500 independent scene trials.

### F.3 New objects, containers, and object counts

The additional generalization setting uses 100 externally designed textured assets: 50 from categories absent in training and 50 new instances from seen categories. Ten previously unseen box-like containers vary in base area and height. Each contains 20–35 selected objects, with reuse permitted and approximately ten unseen-category objects on average. Table 9 retains the reported scores. This tests new assets and regular container dimensions; it does not establish generalization to arbitrary enclosure shapes.

Additional food-storage examples use 20–50 externally designed objects. Table 10 reports quality, and Table 11 reports runtime on an RTX 5090 with 32 GB memory. The total is the sum of planning, refinement, and physical relaxation for each reported size.

### F.4 Reference-pipeline ablations

Table 12 records the original stage-level ablation, while Table 13 records the additional structural comparisons. Their rankings concern their respective candidate sets and should not be compared numerically with ranks in other tables. The tuned-planner row changes training relative to the base model, but the full collection of module changes is not a dataset-size ablation.

In the separately reported direct-pose comparison, direct $R, t$ prediction obtains realism/density/support/placement/semantic scores of 3.79/4.05/3.55/3.78/3.76 and 78.8% physical stability; point prediction with rigid projection obtains 4.10/4.33/3.83/4.03/4.07 and 100%. These observations motivate the reference implementation but do not imply that alternative output representations are fundamentally incapable of modeling storage arrangements.

Table 10: **Varying the input object count.**

| Objects | R | D | S | P | Sem | Physical |
|---|---|---|---|---|---|---|
| 20 | 3.72 | 3.40 | 3.92 | **4.05** | **3.70** | **100%** |
| 30 | 3.85 | 3.72 | 3.85 | 3.98 | 3.68 | **100%** |
| 40 | 3.95 | 3.95 | 3.78 | 3.92 | 3.66 | **100%** |
| 50 | **3.98** | **4.14** | **3.71** | 3.89 | 3.64 | **100%** |

Table 11: **Reported layout runtime.** Stage times and totals are in seconds; memory is in GB.

| Objects | Planner | Refiner | Physics | Total | Memory |
|---|---|---|---|---|---|
| 20 | **11.2** | **64.8** | 2.4 | **78.4** | **10.5** |
| 30 | 12.8 | 96.2 | **2.1** | 111.7 | 16.2 |
| 40 | 14.5 | 127.5 | 2.6 | 144.9 | 22.8 |
| 50 | 16.1 | 158.0 | 2.5 | 177.1 | 29.1 |

## G Single-View Reconstruction: Adaptations and Protocol

TRELLIS.2 (Xiang et al., 2024) receives a single view and can reconstruct a scene-like shape, but the original output does not directly separate the constituent instances in this evaluation. The adaptation segments foreground instances, assigns a distinct color to each, and fine-tunes a diffusion module using InnerScene supervision for one day on eight A100 80 GB GPUs. Color-based clustering separates the resulting mesh for object-level evaluation. Consequently, the adapted model receives instance information absent from the unmodified RGB-only comparison.

SAM 3D (Chen et al., 2025) reconstructs objects and predicts their placement. We assemble its individual predictions into the scene for evaluation. PartCrafter (Lin et al., 2025b) produces a compositional scene from an image and an object count; we fine-tune it on 1,230 scenes for three days on eight A100 80 GB GPUs. The existing task does not enforce correspondence with input 2D masks for every PartCrafter output, so missing, additional, or misplaced geometry remains possible. Figure 5 illustrates the methods under the evaluated adaptations.

Table 3 preserves the numerical values of the existing experiment. CD denotes Chamfer distance and F denotes F-score, evaluated at both scene and object levels. Their exact normalization and threshold conventions have not been supplied in the current implementation record, so we do not assign unverified metric units. The absence of object scores for original TRELLIS.2 reflects the lack of reported instance-separated evaluation, not a measured zero.

## H Robot Placement Demonstrations

Demonstrations use the GALAXEA R1 Lite's right-arm gripper in cleaning-supply and kitchenware cabinets. Each group contains six or seven objects. An operator loads each object before the robot executes the placement segment. No autonomous grasp acquisition or learned policy is included.

Retargeting takes registered hand motion, corresponding object motion, and scene geometry as inputs. These jointly constrain the end effector. Retargeting uses the registered hand and object motion together with the scene geometry to constrain the end effector.

Table 12: **Stage-level layout ablation.** The refiner row precedes final physics relaxation. Scores and ranking summarize the original ablation set.

| Variant | R | D | S | P | Sem | Rank | First | Physical |
|---|---|---|---|---|---|---|---|---|
| Qwen3-VL | 1.13 | 1.63 | 1.13 | 1.13 | 1.13 | 3.75 | 0.0% | 0.0% |
| Tuned planner | 2.00 | 1.13 | 2.75 | 2.50 | 1.88 | 3.25 | 0.0% | 13.2% |
| + Refiner | 4.25 | 4.25 | 4.38 | 4.25 | 4.25 | 1.63 | 37.5% | 84.5% |
| + Physics | **4.75** | **4.63** | **4.63** | **4.75** | **4.75** | **1.38** | **62.5%** | **100%** |

Table 13: **Structural ablations of the reference layout model.** The point-coordinate output is projected to rigid object poses.

| Variant | Average rank | First-place frequency | Physical |
|---|---|---|---|
| Direct SE(3) regression | 3.62 | 8.3% | 78.8% |
| Without support-graph GNN | 4.00 | 5.0% | 56.8% |
| PointNet++ encoder (Qi et al., 2017) | 2.05 | 25.0% | 88.6% |
| Flat transformer | 4.08 | 3.3% | 61.4% |
| Reference pipeline | **1.25** | **58.3%** | **100.0%** |