ROBOPHYSCAN: A LARGE-SCALE SIM-READY REAL-WORLD OBJECT DATASET WITH MANIPULATION ANNOTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robotic manipulation requires object assets that are not only visually realistic, but also sim-ready and reproducible as real-world instances. Existing resources typically provide either scan-derived real-world appearance or simulation metadata, but rarely both for large-scale purchasable objects. Thus, we introduce **RoboPhyScan**, a large-scale high-quality **sim-ready real-world** object dataset that contains 1,079 purchasable everyday object instances across 65 object types. Each asset aligns high-fidelity scanned geometry, reconstructed texture, precise articulation, rich physical properties, and manipulation annotations covering part pose, affordance, and grasp pose. Furthermore, we design an **efficient real-to-sim assetization pipeline** to reduce active human production time to approximately 60 minutes per asset on average. We also systematically evaluate our assets, including visual quality, sim-ready stability, and utility in downstream robotic applications. With RoboPhyScan, we provide a bridge from purchasable real objects to high-quality sim-ready assets, establishing a basis for reproducible sim-real manipulation research.

1 INTRODUCTION

Robotic manipulation is moving beyond controlled laboratories into complex real-world environments (Billard & Kragic, 2019; Spong et al., 2020), placing growing demands on the infrastructure of manipulation policies. Scaling pretrained robotic foundation models requires broad and diverse robot experience (Brohan et al., 2023; Kim et al., 2025; Black et al., 2025b; Team et al., 2025), for which simulation provides a low-cost, scalable complement to expensive physical data collection (Mandlekar et al., 2023; Nasiriany et al., 2024; Wang et al., 2024). Simulation also enables low-cost, repeatable policy evaluation at scale (Liu et al., 2023; Li et al., 2023; Gu et al., 2023) and predicts real-world policy performance (Li et al., 2025; Sedlacek et al., 2026). Yet both data and evaluation capability are bounded by the sim-to-real gap (Tobin et al., 2017; Peng et al., 2018).

Addressing this gap requires assets that make simulation physically reliable while faithfully representing real-world objects. Stable, sim-ready assets enable the collection of complex, contact-rich behaviors at scale (Xiang et al., 2020; Liu et al., 2022; Li et al., 2023; Jin et al., 2026). When these assets are paired with their physical counterparts, simulation and reality can share matched tasks and evaluation protocols, allowing simulated outcomes to be directly validated against real-world performance (Calli et al., 2015; Chao et al., 2021; Li et al., 2025; Sedlacek et al., 2026). High-quality assets with manipulation knowledge also support downstream robot operation in motion-planning pipelines (Sundaralingam et al., 2023; Curtis et al., 2022). More broadly, high-fidelity assets provide the executable substrate needed to transfer policies trained in simulation to the real world (Villasevil et al., 2024; Han et al., 2026).

These requirements motivate a large-scale sim-ready real-world object dataset, yet existing resources leave a clear gap. Scan-based real-world and digital-twin datasets provide faithful geometry and appearance (Calli et al., 2015; Downs et al., 2022; Wu et al., 2023; Kuang et al., 2023; Dong et al., 2025), but the availability of articulation, physical metadata, and manipulation annotations varies across these resources. GSO, for example, releases rigid-body simulation models with mass and inertia estimates and collision geometry (Downs et al., 2022); our focus is on combining real-object

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.

Table 1: Fine-grained comparison with representative real-world or sim-ready datasets. 'Y' indicates that a capability is released systematically, 'P' indicates partial or subset-level support, and 'N' indicates that it is not a central released component.

	Dataset Identity			Appearance		Sim-Ready					Manipulation Knowledge			Reproducibility
Dataset	Real	#Objs	#Cats	Texture	PBR	Part Seg.	Joint	Collision	Physics	Modular	Afford.	Part Pose	Grasp Pose	Purchasable
GSO (Downs et al., 2022)	Y	1030	17	Y	N	N	N	Y	P	N	N	N	N	P
OmniObject3D (Wu et al., 2023)	Y	6000	190	Y	N	N	N	N	N	N	N	N	N	N
Stanford-ORB (Kuang et al., 2023)	Y	14	–	Y	Y	N	N	N	N	N	N	N	N	N
Digital Twin Catalog (Dong et al., 2025)	Y	2000	40	Y	Y	N	N	N	N	N	N	N	N	P
YCB (Calli et al., 2015)	Y	77	5	P	N	N	N	P	P	N	N	N	N	Y
PartNet-Mobility (Xiang et al., 2020)	N	2346	46	Y	N	Y	Y	Y	Y	N	N	N	N	N
AKB-48 (Liu et al., 2022)	Y	2037	48	Y	N	P	Y	Y	Y	N	N	P	N	N
GAPartNet (Geng et al., 2023)	P	1166	27	Y	N	P	Y	Y	P	N	N	P	N	N
BEHAVIOR-1K (Li et al., 2023)	P	9000+	1900+	Y	Y	P	Y	Y	Y	P	P	N	N	N
RoboCasa (Nasiriany et al., 2024)	N	2509	153	Y	Y	P	P	Y	Y	P	P	N	N	N
ArtVIP (Jin et al., 2026)	P	992	9	Y	Y	Y	Y	Y	P	Y	Y	N	N	P
RoboPhyScan (ours)	**Y**	**1079**	**65**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**

appearance with articulated interaction and manipulation annotations. Conversely, articulated and simulation-oriented resources provide part structure, joints, and interaction behaviors (Xiang et al., 2020; Liu et al., 2022; Geng et al., 2023; Li et al., 2023), but their assets are often synthetic, manually authored, or limited in geometric and appearance quality, leaving a substantial gap from real-world objects.

To close this gap, we introduce **RoboPhyScan**, a sim-ready real-world object dataset with manipulation knowledge, together with an efficient real-to-sim assetization pipeline. The dataset pairs purchasable objects with scanned geometry, PBR appearance, articulation, physical properties, and manipulation knowledge annotations. To make this construction process scalable and reproducible, our pipeline standardizes object reconstruction, concept-based annotation, and simulator-level verification into a repeatable production workflow.

In conclusion, our key contributions are as follows: **(1) A Large-scale Sim-Ready Real-World Object Dataset with Manipulation Annotations.** We provide 1,079 everyday object instances across 65 object types, each paired with a purchasable real-world instance and released with geometry, appearance, articulation, physics, and manipulation labels. **(2) An Efficient Real-to-Sim Assetization Pipeline.** We standardize object capture, reconstruction, concept-based annotation, and simulator-level verification into a repeatable production workflow. **(3) A Systematic Evaluation for the Quality of Sim-Ready Real-World Objects.** We evaluate visual fidelity, simulator executability, contact stability, and manipulation-knowledge consistency under unified protocols, and demonstrate downstream utility through motion planning and policy learning.

2 Related Works

Table 1 compares RoboPhyScan with representative real-world or sim-ready datasets. RoboPhyScan uniquely provides real-world purchasable objects, high-fidelity appearance, simulator-ready articulation and physics, and manipulation-oriented annotations within the same per-object package.

2.1 Real-World Object Datasets

YCB (Calli et al., 2015) is a classic effort in purchasable real-world assets, but it contains only 77 rigid objects. Google Scanned Objects (Downs et al., 2022) contributed 1,030 household scans with single-link SDF simulation models, collision geometry, and measured or estimated mass and geometry-derived inertia. Friction is left to simulator defaults, and the models do not include articulation. OmniObject3D (Wu et al., 2023) scaled scanned real-world object coverage to 6,000 objects across 190 categories. Stanford-ORB (Kuang et al., 2023) focuses on real-world inverse rendering with scans, multi-view images, and lighting, while Digital Twin Catalog (Dong et al., 2025) emphasizes photorealistic object digital twins with high-quality geometry and PBR materials. These resources differ in their simulation support; our focus is on jointly providing real-object appearance, articulated structure, physical metadata, and manipulation annotations.



Figure 1: Representative RoboPhyScan objects.

2.2 Sim-Ready Object Datasets

PartNet-Mobility (Xiang et al., 2020) adds articulated kinematic structure to CAD objects for interactive simulation. BEHAVIOR-1K (Li et al., 2023) collects high-quality handcrafted assets tied to its simulation ecosystem. RoboCasa (Nasiriany et al., 2024) scales household simulation assets for generalist robots, but only a small subset of its assets are articulated and the resource is also not a real-world collection. AKB-48 (Liu et al., 2022) scans real articulated objects and organizes an articulation knowledge graph, but its asset quality and purchasability remain limited. ArtVIP (Jin et al., 2026) offers high-quality articulated assets with modular interaction and affordance annotations, but it is also handcrafted, leaving a gap in the sim-real and asset reproducibility.

2.3 Real-to-Sim and Asset Construction Pipelines

Existing pipelines focus on digital-twin reconstruction (Downs et al., 2022; Wu et al., 2023; Kuang et al., 2023; Dong et al., 2025), articulated asset authoring (Liu et al., 2022; Xiang et al., 2020; Li et al., 2023; Nasiriany et al., 2024; Jin et al., 2026), or real-to-sim task generation (Villasevil et al., 2024; Han et al., 2026), but typically prioritize visual reconstruction, simulator-authored models, or task-specific scenes and trajectories, respectively. RoboPhyScan connects these parts through a systematic real-to-sim assetization pipeline that converts purchasable real objects into reusable, sim-ready asset packages with scanned geometry and appearance, articulation, physics, and manipulation annotations.

3 RoboPhyScan Dataset Design and Construction Pipeline

3.1 Object Sourcing and Dataset Characteristics

RoboPhyScan targets common objects that appear frequently in robotic interaction settings and daily life, with representative instances shown in Figure 1. The dataset contains 1,079 object instances from 65 object types. We select objects that are interactive and easy to purchase, while keeping intra-type shape diversity and broad coverage of common manipulation modes. Each released asset retains a purchase link. Under the current single-operator workflow, the pipeline requires about 60 minutes of active human production time per validated asset on average.

3.2 Geometry Acquisition and Post-processing

We acquire geometry with a SHINING 3D EinScan HX2 handheld scanner[1] using the controlled capture configuration shown in Figure 5(a). Each object is scanned in at least two poses and for at least one complete turntable revolution per pose. The partial scans are then registered and fused

[1] https://www.shining3d.com

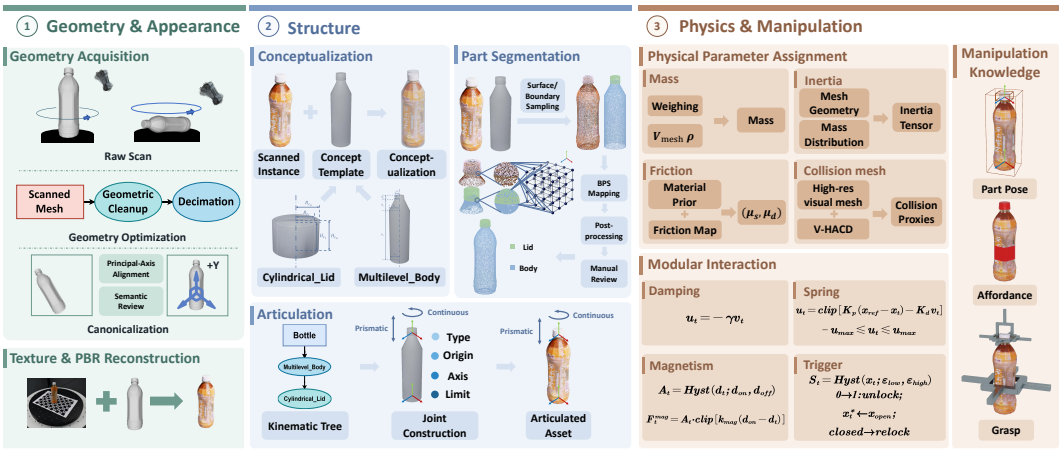

Figure 2: Overview of our real-to-sim assetization pipeline. Our holistic workflow transforms real-world objects into high-fidelity, sim-ready articulated assets.

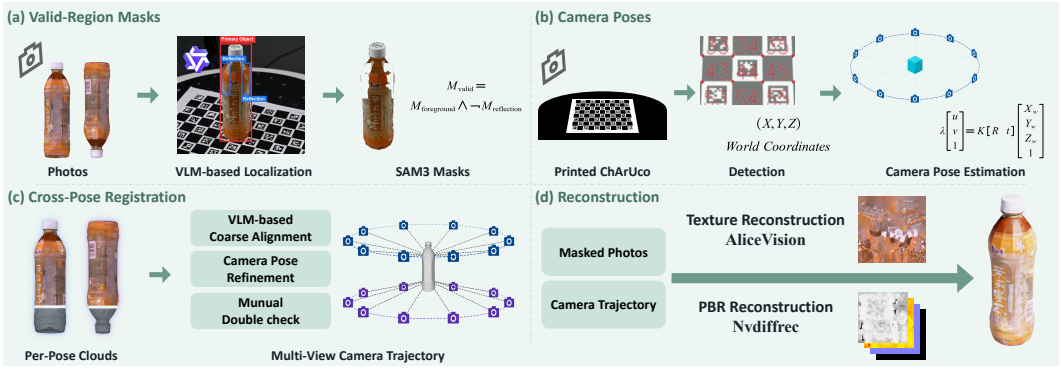

Figure 3: Overview of our texture and PBR reconstruction pipeline.

into a complete mesh. Under our current single-operator, single-scanner setting, it requires approximately 25–30 minutes of active human time per object.

Raw scans commonly contain millions to tens of millions of faces, together with disconnected fragments, overlapping geometry, self-intersections, non-manifold structures, and holes, making them unsuitable for direct downstream use. We first remove invalid geometry using MeshLab (Cignoni et al., 2008) and PyMeshLab (Muntoni & Cignoni, 2021), and then apply Screened Poisson Surface Reconstruction (Kazhdan & Hoppe, 2013) to recover a smooth and topologically regularized surface. The reconstructed visual mesh is standardized to a target of approximately 300K faces through quadric edge-collapse decimation (Garland & Heckbert, 1997). This target represents an empirical trade-off between retaining local geometric detail and reducing the computational, storage, and loading costs of subsequent processing. We then apply cotangent-weighted Laplacian coordinate smoothing (Sorkine, 2005) to suppress local jaggedness, followed by degenerate-face removal and normal recomputation to obtain a continuous and topologically stable mesh.

After geometric cleaning, we align each model's principal inertia axes with the canonical axes and translate its bounding-box center to the origin, using a right-handed coordinate system with $+Y$ as up. Annotators then resolve the semantic axis directions through rigid rotations without reflections. Objects that cannot be oriented reliably from their principal axes are refined manually in CloudCompare (CloudCompare, n.d.).

3.3 Texture and PBR Reconstruction

Using the appearance-acquisition configuration in Figure 5(b), we photograph each object in multiple poses to cover all texture. It takes approximately 5 minutes of active human time per object. For each object pose, we independently estimate camera extrinsics using the ChArUco implementation in OpenCV (Bradski, 2000)[2]. To suppress background leakage and unstable specular regions during texture fusion, we utilize Qwen3.7-Plus-Thinking and SAM3 (Carion et al., 2026) to localize and mask the reflective areas to be ignored.

Images and camera parameters captured under different object poses initially lie in independent local coordinate frames. For each pose, we construct the corresponding COLMAP camera representation (Schonberger & Frahm, 2016) and a proxy point cloud. We register the proxy to the high-precision mesh using CloudCompare (CloudCompare, n.d.), with assistance from GPT-5.5 or manual registration by a professional engineer for more complex objects, and apply the resulting transformation to the associated cameras. This yields a unified multi-view camera trajectory for texture processing. We then refine the camera extrinsics using mutual-information (MI) optimization in MeshLab (Cignoni et al., 2008). The processed multi-pose images and refined camera parameters are subsequently supplied to AliceVision (Griwodz et al., 2021) for multi-view color fusion using multiband blending, while preserving the prepared mesh and its UV parameterization. The resulting textured mesh is expressed in the canonical coordinate frame. Under the same setting, we further run nvdiffrec (Munkberg et al., 2022) to optimize PBR material maps. For most objects, this processing workflow requires little to no manual intervention.



Figure 4: Overview of Mesh and Texture Quality.

3.4 Part Segmentation and Articulation

Part segmentation is a prerequisite for articulation, but manual precise mesh splitting is difficult to scale. We therefore design a human-in-the-loop segmentation workflow. Each scanned object is aligned with its analytic template (Sun et al., 2024), whose part semantics are transferred to the asset mesh faces using BPS (Prokudin et al., 2019). We obtain initial face labels from complementary global, local, and surface-normal cues, then refine them using template-derived geometric support and connectivity-aware boundary cleanup. The result is inspected in Autodesk 3ds Max[3], where annotators correct remaining boundary errors and merge redundant fragments.

After segmentation, each object is converted into an articulated asset through the same concept-based propagation principle. We mathematically define category-level kinematic structure and joint parameters on concept templates, which we then propagate to each segmented part to instantiate its articulation. We then adapt joint placement, link geometry, collision support, and motion limits to the individual object. We support basic articulation and composite decompositions for mechanisms whose real motion combines rotation and translation, such as the threaded bottle-neck and cap assemblies detailed in Appendix G. It takes approximately 12 minutes of active human time per object to conceptualize one object and 8 minutes to correct fine-grained segmentation errors.

[2] https://docs.opencv.org/4.x/df/d4a/tutorial_charuco_detection.html
[3] https://www.autodesk.com/products/3ds-max/overview

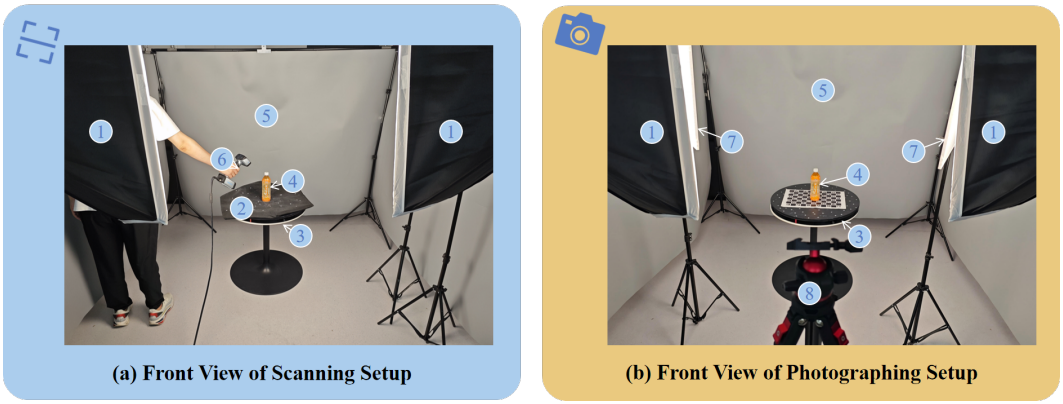

Figure 5: Capture setups used for geometry scanning and appearance acquisition. (a) Geometry is captured under suppressed ambient illumination with the softboxes operated at low power. (b) Multi-view photographs are captured with stronger diffused illumination and a fixed camera. The numbered labels denote (1) softboxes, (2) high-contrast tablecloth, (3) motorized turntable, (4) target object, (5) diffuse gray PVC backdrop, (6) handheld scanner, (7) diffusion sheets, and (8) camera.

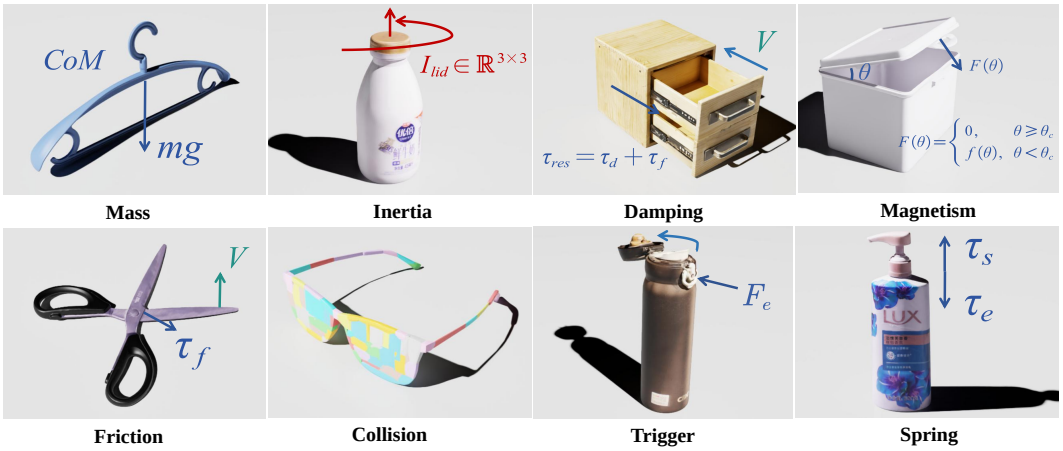

Figure 6: Illustration of physical properties and interaction behaviors represented in RoboPhyScan. RoboPhyScan augments visual meshes with executable physical parameters categorized into two primary groups: (1) Fundamental Rigid-Body Properties (left two columns): basic attributes for dynamic simulation. (2) Modular Interaction (right two columns): higher-order interaction behaviors commonly required for articulated objects.

3.5 Physical Parameter Assignment

RoboPhyScan narrows the gap between visual realism and physical executability by attaching physical parameters at the asset level. Mass is measured directly with a standardized digital scale when the object's constituent parts can be disassembled and weighed independently; otherwise, it is estimated from mesh volume combined with material-density priors. Inertia tensors are computed from the cleaned geometry and assigned mass distribution using Trimesh (Dawson-Haggerty, n.d.). Friction parameters are obtained by mapping material priors inferred by GPT-5.5 to material-specific friction coefficients. Static and dynamic coefficients are stored separately and subjected to relative random perturbations of up to 10% to account for estimation error and real-world variation within the same material category.

RoboPhyScan also supports higher-order interaction behaviors commonly required for articulated assets, including joint damping for hinges and sliders, spring-like return for elastic switches or self-closing parts, magnetic closure for snap-to-fit components, and trigger-like coupling. Collision meshes are generated separately from visual geometry using V-HACD (Mamou, 2016), allowing

Table 2: Object and part mesh quality. Tri. med.: Triangle Count Median; Bnd.: Boundary Edges; Self-int.: Self-intersections. DTC has no part-level meshes.

Dataset	Tri. med. (k)		Bnd. (%) ↓		Self-int. (%) ↓	
	Object	Part	Object	Part	Object	Part
RoboPhyScan	299.9	81.5	**0.049**	**0.000**	**0.001**	**0.008**
DTC	249.5	–	3.611	–	4.388	–
AKB-48	184.8	45.1	2.549	3.971	4.064	4.396
PartNet-Mobility	6.7	1.1	**0.000**	54.50	83.91	68.57

high-resolution visual meshes to coexist with efficient contact computation. Each exported asset is verified in simulation to ensure stability and executability before release.

3.6 Manipulation Annotation

Following ConceptFactory (Sun et al., 2024), we first define object-centric knowledge mathematically on reusable concept templates and propagate the knowledge to the object instance. The released knowledge includes part pose, affordance, and grasp pose. This conceptualization and knowledge-propagation design allows other object-centric manipulation labels to be added in the same way by defining them on the corresponding concept templates.

4 Evaluation and Application

4.1 Geometry and Appearance Quality

We assess the geometry and appearance quality of the RoboPhyScan assets. Mesh topology quality evaluates whether the reconstructed geometry is structurally valid and suitable for downstream processing. Texture quality measures how faithfully the reconstructed appearance preserves that of the real-world object. Together, these evaluations separate structural validity, appearance fidelity, and geometric accuracy.

Mesh Quality. We evaluate object and part mesh quality using triangle count, boundary edges, and self-intersections for RoboPhyScan, AKB-48 (Liu et al., 2022), PartNet-Mobility (Xiang et al., 2020), DTC (Dong et al., 2025). Triangle count characterizes mesh complexity and geometric accuracy. Boundary-edge ratios quantify open mesh boundaries, while self-intersection ratios quantify surface interpenetration. Detailed definitions are provided in Appendix D.1. As shown in Table 2, our object and part meshes preserve details and present high quality with nearly zero boundary edges and self-intersections.

Texture Quality. We randomly select 200 objects and photograph each from eight views with a calibrated camera. Each photograph is independently provided as a single-image input to SAM3D (Chen et al., 2026) and Hunyuan3D2.1 (Team, 2025), yielding one textured model per method and view. For each object–view pair, we render the corresponding single-image reconstructions and our object asset from the captured camera view, aligning the assets using the pose-matching procedure in Sec. 3.3. Each rendering is compared with its corresponding real photograph.

Table 3: Texture-fidelity comparison.

Dataset	LPIPS↓	DISTS↓	CLIP↑	DINOv2↑
RoboPhyScan	**0.3540**	**0.1945**	**0.9106**	**0.7924**
SAM3D	0.4211	0.2819	0.8587	0.6623
Hunyuan3D2.1	0.3804	0.2634	0.8463	0.6426

We quantify the agreement between each rendered image and its corresponding real photograph using four complementary metrics. LPIPS (Zhang et al., 2018) and DISTS (Ding et al., 2022) measure perceptual distance, while CLIP (Radford et al., 2021) and DINOv2 (Oquab et al., 2024) measure cosine similarity in semantic and visual representation spaces. As shown in table 3, RoboPhyScan presents higher consistency between the rendered appearance and the real object image.

Table 4: Unified sim-ready asset quality evaluation across datasets.

Dataset	Static	Settling	Push	Articulation	Grasp	Actuation
RoboPhyScan	**89.5**%	**86.0**%	**91.0**%	**89.0**%	**92.5**%	**72.5**%
ArtVIP	75.0%	82.5%	86.0%	72.5%	80.0%	64.5%
PartNet-Mobility	65.0%	63.5%	67.5%	59.0%	60.5%	49.5%

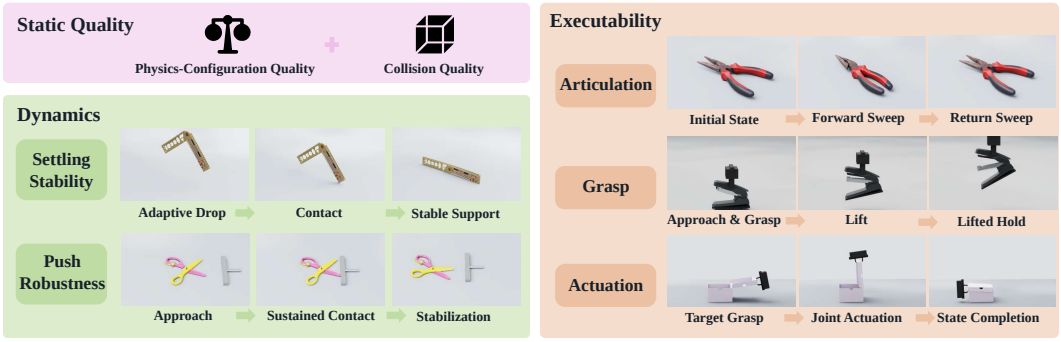

Figure 7: Overview of the sim-ready validation protocol.

4.2 Sim-Ready Quality

We focus this subsection on whether the sim-ready assets are operationally ready for simulation under a unified validation protocol. We evaluate 200 articulated objects each from RoboPhyScan, ArtVIP (Jin et al., 2026), and PartNet-Mobility (Xiang et al., 2020). For PartNet-Mobility and ArtVIP, available grasp and affordance information is attached using the same concept-template framework as RoboPhyScan. All assets are evaluated in Isaac Sim 5.1.0 (NVIDIA, 2025). As shown in Figure 7, our protocol evaluates six complementary aspects of sim-ready quality. Table 4 reports the pass rates: the fraction of all 200 objects per dataset that pass each metric.

Static Quality. We inspect physical configuration and collision geometry. Passing requires both configuration plausibility and collision checks for fidelity, integrity, and initial overlap to pass.

Settling Stability. We release each asset above the ground. A trial passes when ground contact and support are established, with drift, residual speed, and penetration below their thresholds.

Push Robustness. We push each asset from four horizontal directions. A trial passes when at least three directions achieve contact and stable recovery without penetration or unstable motion.

Articulation Quality. We drive finite-range joints forward and back. A trial passes when all tested joints meet completion thresholds in both directions without limit violations or unstable motion.

Grasp Quality. We test lift-and-hold grasps. Each gripper geometry must pass contact, ground-clearance, slip, motion, and penetration checks.

Actuation Quality. We grasp and drive a movable part. One directional attempt must achieve sufficient contact and progress within joint and velocity limits.

Appendix E details the pass criteria and trial aggregation. RoboPhyScan achieves the highest pass rates across all six dimensions, indicating more consistent physical configuration and collision validity, stability under settling and pushing, and manipulation executability.

4.3 Downstream Application

We demonstrate the practical utility of RoboPhyScan through two downstream applications: asset-grounded motion planning and policy learning from simulated demonstrations. We study four tasks—TwistCap, PressShampoo, InsertUSB, and OpenStapler—that require accurate geometry, manipulation annotations, and stable physical behavior. For each configuration, we conduct 20

Table 5: Real-world task success rates. 'R-n' and 'S-n' denote n real and simulated training demonstrations, respectively, and combined settings use both sources.

Method	Data	TwistCap	PressShampoo	InsertUSB	OpenStapler
cuRobo	–	95%	100%	95%	100%
DP	R-150	20%	40%	25%	40%
	S-150	10%	25%	10%	25%
	R-150+S-75	30%	55%	35%	55%
	R-150+S-150	45%	60%	40%	70%
$\pi_{0.5}$	R-150	35%	60%	40%	55%
	S-150	25%	45%	25%	40%
	R-150+S-75	45%	70%	50%	65%
	R-150+S-150	55%	80%	65%	80%

trials on the real robot for each task and report the resulting success rates in Table 5. All simulations run in Isaac Sim 5.1.0 (NVIDIA, 2025), with detailed settings provided in Appendix F.

Experimental Setup. Real-world experiments use a Flexiv Rizon 4 robot equipped with a Flexiv Grav GN-01 gripper, a wrist-mounted Intel RealSense D435i camera, and an external Intel RealSense D455 camera. We use plain black and green backdrops to minimize scene variation and maintain consistent visual conditions across data collection and evaluation. Table 5 reports real-world success rates for all four tasks, with one result per task and method setting.

Asset-Grounded Motion Planning. FoundationPose (Wen et al., 2024) estimates the object or part pose using the corresponding asset. The estimated pose maps manipulation annotations—including affordance regions, grasp poses, and keypoints—from the asset frame to the robot world frame. We instantiate waypoints from this grounded knowledge and use cuRobo (Sundaralingam et al., 2023) for motion planning.

Imitation Learning and VLA. This application uses demonstrations generated with RoboPhyScan assets for imitation learning and VLA policy post-training. For data collection, we use our motion-planning pipeline in both Isaac Sim and the real robot setup, yielding 150 simulated and 150 real successful trajectories. We use Diffusion Policy (DP) (Chi et al., 2023) to measure the benefit for task-specific policies trained from scratch, and $\pi_{0.5}$ (Black et al., 2025a) to measure whether the same simulated executions can help adapt a pretrained VLA model. Policies are trained with real-only (R-150), simulation-only (S-150), and mixed demonstrations (R-150+S-75 and R-150+S-150), where R and S denote real and simulated training trajectories, respectively. The R-150+S-75 setting measures the effect of adding a smaller simulated-data budget while keeping the full real-data budget fixed. All four settings are evaluated on the real robot.

Discussion. Table 5 summarizes our success rates across methods and presents three findings. First, high-fidelity assets and manipulation knowledge support reliable real-world motion plans. Second, simulation-only demonstrations produce non-zero real-world success, showing that the assets preserve task-relevant structure across the sim-to-real transition. Third, adding simulation data helps improve performance over real-only training for both policies. These results indicate that RoboPhyScan assets can help narrow the sim-to-real gap for downstream manipulation.

5 Conclusion

We present RoboPhyScan, a large-scale sim-ready real-world object dataset that bridges high-fidelity scanning and executable robotic interaction. Across our systematic evaluation and downstream applications, RoboPhyScan demonstrates strong visual fidelity, robust simulator executability, and practical manipulation utility, showing that real-world object data can be organized as executable infrastructure.

AI USE STATEMENT

This work used generative AI tools for several explicitly supervised research and writing tasks. GPT-5.5 was used to improve the clarity and readability of the manuscript, provide feedback on methodological exposition and experimental reporting, assist with camera-to-mesh registration in selected complex cases (Sec. 3.3), and infer material labels used to initialize physical-parameter assignment (Sec. 3.5). Qwen3.7-Plus-Thinking and SAM3 were used to assist with reflective-region localization and masking during texture processing. These model-assisted outputs were reviewed by the authors; material labels were manually verified, registration results were checked, and all exported assets were subjected to simulator-level validation. The authors take responsibility for the final manuscript, claims, data, code, and artifacts.

REPRODUCIBILITY STATEMENT

The paper describes the complete asset-construction workflow, including object sourcing, geometry acquisition, texture and PBR reconstruction, part segmentation, articulation, physical parameter assignment, and manipulation annotation (Sec. 3). Appendix C provides the mathematical construction of physical properties and modular interactions; Appendices D and E specify the quality metrics, trial structure, and aggregation procedures; and Appendix F gives the downstream manipulation and training-data protocols. The manipulation-knowledge representation and exporter examples are included in Appendix G. Anonymous reviewer-facing links to example assets, pipeline code, and demonstration videos are provided in Appendix A, together with compute-resource information and release documentation.

ETHICS STATEMENT

RoboPhyScan contains scans and metadata for purchasable everyday objects and does not contain personally identifiable information, human-subject data, or offensive content. Human labor was used for object capture, mesh cleanup, orientation verification, texture alignment, part segmentation, concept fitting, simulator export, and quality control; these activities were treated as compensated data-curation and asset-production work, with task instructions and quality criteria provided to workers. Existing datasets, software, models, and third-party assets are used under their respective licenses, and released RoboPhyScan assets include provenance, licensing information, and known caveats. The reported simulator validation measures asset executability and does not constitute a guarantee of safe deployment on a physical robot.

REFERENCES

Aude Billard and Danica Kragic. Trends and challenges in robot manipulation. *Science*, 364(6446): eaat8414, 2019. doi: 10.1126/science.aat8414. URL https://www.science.org/doi/abs/10.1126/science.aat8414.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, brian ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 27–30 Sep 2025a. URL https://proceedings.mlr.press/v305/black25a.html.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control.

In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025b. doi: 10.15607/RSS.2025.XXI.010.

G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S Ryoo, Grecia Salazar, Pannag R Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan H Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-1: Robotics Transformer for Real-World Control at Scale. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.025.

Berk Calli, Arjun Singh, Aaron Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M. Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 International Conference on Advanced Robotics (ICAR)*, pp. 510–517, 2015. doi: 10.1109/ICAR.2015.7251504.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, RISHI HAZRA, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollar, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 138846–138923, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/e0982cbc81401df3430ee1ff780dc7a2-Paper-Conference.pdf.

Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S. Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, Jan Kautz, and Dieter Fox. Dexycb: A benchmark for capturing hand grasping of objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9044–9053, June 2021.

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

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin CM Burchfiel, and Shuran Song. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.026.

Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, and Guido Ranzuglia. MeshLab: an Open-Source Mesh Processing Tool. In Vittorio Scarano, Rosario De Chiara, and Ugo Erra (eds.), *Eurographics Italian Chapter Conference*. The Eurographics Association, 2008. ISBN 978-3-905673-68-5. doi: 10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2008/129-136.

CloudCompare. CloudCompare, n.d. URL https://www.cloudcompare.org/. 3D point-cloud and mesh-processing software. Accessed 2026-09-25.

Aidan Curtis, Xiaolin Fang, Leslie Pack Kaelbling, Tomás Lozano-Pérez, and Caelan Reed Garrett. Long-horizon manipulation of unknown objects via task and motion planning with estimated

affordances. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 1940–1946, 2022. doi: 10.1109/ICRA46639.2022.9812057.

Michael Dawson-Haggerty. trimesh, n.d. URL https://trimesh.org/. Python mesh-processing library. Accessed 2026-09-25.

Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, 2022. doi: 10.1109/TPAMI.2020.3045810.

Zhao Dong, Ka Chen, Zhaoyang Lv, Hong-Xing Yu, Yunzhi Zhang, Cheng Zhang, Yufeng Zhu, Stephen Tian, Zhengqin Li, Geordie Moffatt, Sean Christofferson, James Fort, Xiaqing Pan, Mingfei Yan, Jiajun Wu, Carl Yuheng Ren, and Richard Newcombe. Digital twin catalog: A large-scale photorealistic 3d object digital twin dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 753–763, June 2025.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560, 2022. doi: 10.1109/ICRA46639.2022.9811809.

Michael Garland and Paul S. Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pp. 209–216, USA, 1997. ACM Press/Addison-Wesley Publishing Co. ISBN 0897918967. doi: 10.1145/258734.258849. URL https://doi.org/10.1145/258734.258849.

Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7081–7091, June 2023.

Carsten Griwodz, Simone Gasparini, Lilian Calvet, Pierre Gurdjos, Fabien Castan, Benoit Maujean, Gregoire De Lillo, and Yann Lanthony. Alicevision Meshroom: An open-source 3D reconstruction pipeline. In *Proceedings of the 12th ACM Multimedia Systems Conference - MMSys '21*. ACM Press, 2021. doi: 10.1145/3458305.3478443.

Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. Maniskill2: A unified benchmark for generalizable manipulation skills. In *International Conference on Learning Representations*, 2023.

Xiaoshen Han, Junqiu Yu, Minghuan Liu, Yilun Chen, Xiaoyang Lyu, Yang Tian, Bolun Wang, Weinan Zhang, Weinan Zhang, and Jiangmiao Pang. Re^{3}sim: Generating high-fidelity simulation data via 3d-photorealistic real-to-sim for robotic manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

Zhao Jin, Zhengping Che, Tao Li, Zhen Zhao, Kun Wu, Yuheng Zhang, Yinuo Zhao, Zehui Liu, Qiang Zhang, Xiaozhu Ju, Jing Tian, Yousong Xue, and Jian Tang. Artvip: Articulated digital assets of visual realism, modular interaction, and physical fidelity for robot learning. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 75272–75301, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/7a03c2bf486e56d8a25e8d5bb72ff1a2-Paper-Conference.pdf.

Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Trans. Graph.*, 32(3), July 2013. ISSN 0730-0301. doi: 10.1145/2487228.2487237. URL https://doi.org/10.1145/2487228.2487237.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. In Pulkit Agrawal, Oliver Kroemer, and

Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/kim25c.html.

Zhengfei Kuang, Yunzhi Zhang, Hong-Xing Yu, Samir Agarwala, Elliott / Shangzhe Wu, and Jiajun Wu. Stanford-orb: A real-world 3d object inverse rendering benchmark. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 46938–46957. Curran Associates, Inc., 2023. doi: 10.52202/075280-2033. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/92a821f6c25b29241df6985ceb673a85-Paper-Datasets_and_Benchmarks.pdf.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, Mona Anvari, Minjune Hwang, Manasi Sharma, Arman Aydin, Dhruva Bansal, Samuel Hunter, Kyu-Young Kim, Alan Lou, Caleb R Matthews, Ivan Villa-Renteria, Jerry Huayang Tang, Claire Tang, Fei Xia, Silvio Savarese, Hyowon Gweon, Karen Liu, Jiajun Wu, and Li Fei-Fei. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In Karen Liu, Dana Kulic, and Jeff Ichnowski (eds.), *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 80–93. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/li23a.html.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Oier Mees, Karl Pertsch, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 3705–3728. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/li25c.html.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791. Curran Associates, Inc., 2023. doi: 10.52202/075280-1939. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/8c3c666820ea055a77726d66fc7d447f-Paper-Datasets_and_Benchmarks.pdf.

Liu Liu, Wenqiang Xu, Haoyuan Fu, Sucheng Qian, Qiaojun Yu, Yang Han, and Cewu Lu. Akb-48: A real-world articulated object knowledge base. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14809–14818, June 2022.

Khaled Mamou. Volumetric hierarchical approximate convex decomposition. In Eric Lengyel (ed.), *Game Engine Gems 3*, pp. 141–158. A K Peters, 2016.

Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1820–1864. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/mandlekar23a.html.

Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8280–8290, June 2022.

Alessandro Muntoni and Paolo Cignoni. PyMeshLab, January 2021.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-Scale Simulation of Household Tasks for Generalist Robots. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.050.

NVIDIA. Isaac Sim, 2025. URL https://docs.isaacsim.omniverse.nvidia.com/5.1.0/index.html. Version 5.1.0. Robotics simulation software. Accessed 2026-09-25.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8, May 2018. doi: 10.1109/ICRA.2018.8460528.

Sergey Prokudin, Christoph Lassner, and Javier Romero. Efficient learning on point clouds with basis point sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

Martin Sedlacek, Pavlo Yefanov, Georgy Ponimatkin, Jai Bardhan, Simon Pilc, Mederic Fourmy, Evangelos Kazakos, Cees G. M. Snoek, Josef Sivic, and Vladimir Petrik. Realm: A real-to-sim validated benchmark for generalization in robotic manipulation. *IEEE Robotics and Automation Letters*, 11(7):8315–8322, 2026. doi: 10.1109/LRA.2026.3692088.

Olga Sorkine. Laplacian Mesh Processing. In Yiorgos Chrysanthou and Marcus Magnor (eds.), *Eurographics 2005 - State of the Art Reports*. The Eurographics Association, 2005. doi: 10.2312/egst.20051044.

Mark W. Spong, Seth Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, 2 edition, 2020. ISBN 978-1-119-52399-4. URL https://www.wiley.com/en-us/robot-modeling-and-control-2nd-edition-p-9781119523994.

Jianhua Sun, Yuxuan Li, Longfei Xu, Nange Wang, Jiude Wei, Yining Zhang, and Cewu Lu. Conceptfactory: Facilitate 3d object knowledge annotation with object conceptualization. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 75454–75467. Curran Associates, Inc., 2024. doi: 10.52202/079017-2402. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/89d19544d314740d11c0974ca3ddaf70-Paper-Datasets_and_Benchmarks_Track.pdf.

Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, Nathan Ratliff, and Dieter Fox. Curobo: Parallelized collision-free robot motion generation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8112–8119, 2023. doi: 10.1109/ICRA48891.2023.10160765.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, Steven Bohez, Konstantinos Bousmalis, Anthony Brohan, Thomas Buschmann, Arunkumar Byravan, Serkan Cabi, Ken Caluwaerts, Federico Casarini, Oscar Chang, Jose Enrique Chen, Xi Chen, Hao-Tien Lewis Chiang, Krzysztof Choromanski, David D'Ambrosio,

Sudeep Dasari, Todor Davchev, Coline Devin, Norman Di Palo, Tianli Ding, Adil Dostmohamed, Danny Driess, Yilun Du, Debidatta Dwibedi, Michael Elabd, Claudio Fantacci, Cody Fong, Erik Frey, Chuyuan Fu, Marissa Giustina, Keerthana Gopalakrishnan, Laura Graesser, Leonard Hasenclever, Nicolas Heess, Brandon Hernaez, Alexander Herzog, R. Alex Hofer, Jan Humplik, Atil Iscen, Mithun George Jacob, Deepali Jain, Ryan Julian, Dmitry Kalashnikov, M. Emre Karagozler, Stefani Karp, Chase Kew, Jerad Kirkland, Sean Kirmani, Yuheng Kuang, Thomas Lampe, Antoine Laurens, Isabel Leal, Alex X. Lee, Tsang-Wei Edward Lee, Jacky Liang, Yixin Lin, Sharath Maddineni, Anirudha Majumdar, Assaf Hurwitz Michaely, Robert Moreno, Michael Neunert, Francesco Nori, Carolina Parada, Emilio Parisotto, Peter Pastor, Acorn Pooley, Kanishka Rao, Krista Reymann, Dorsa Sadigh, Stefano Saliceti, Pannag Sanketi, Pierre Sermanet, Dhruv Shah, Mohit Sharma, Kathryn Shea, Charles Shu, Vikas Sindhwani, Sumeet Singh, Radu Soricut, Jost Tobias Springenberg, Rachel Sterneck, Razvan Surdulescu, Jie Tan, Jonathan Tompson, Vincent Vanhoucke, Jake Varley, Grace Vesom, Giulia Vezzani, Oriol Vinyals, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Fei Xia, Ted Xiao, Annie Xie, Jinyu Xie, Peng Xu, Sichun Xu, Ying Xu, Zhuo Xu, Yuxiang Yang, Rui Yao, Sergey Yaroshenko, Wenhao Yu, Wentao Yuan, Jingwei Zhang, Tingnan Zhang, Allan Zhou, and Yuxiang Zhou. Gemini robotics: Bringing ai into the physical world, 2025. URL https://arxiv.org/abs/2503.20020.

Tencent Hunyuan3D Team. Hunyuan3d 2.1: From images to high-fidelity 3d assets with production-ready pbr material, 2025.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017. doi: 10.1109/IROS.2017.8202133.

Marcel Torne Villasevil, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling Reality through Simulation: A Real-To-Sim-to-Real Approach for Robust Manipulation. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.015.

Yufei Wang, Zhou Xian, Feng Chen, Tsun-Hsuan Wang, Yian Wang, Katerina Fragkiadaki, Zackory Erickson, David Held, and Chuang Gan. RoboGen: Towards unleashing infinite data for automated robot learning via generative simulation. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 51936–51983. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/wang24cc.html.

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 (CVPR)*, pp. 17868–17879, June 2024.

Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, Dahua Lin, and Ziwei Liu. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 803–814, June 2023.

Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.