Scalable Trajectory Generation for Whole-Body Mobile Manipulation

Yida Niu [1,3,4,5*] Xinhai Chang [1,3,4,5,6*] Xin Liu [4*] Ziyuan Jiao [2,4✉] Yixin Zhu [3,4,5,7✉]

[1] Institute for AI, Peking University [2] Institute of Unmanned Systems, Beihang University
[3] School of Psychological and Cognitive Sciences, Peking University [4] State Key Laboratory of General Artificial Intelligence
[5] Beijing Key Laboratory of Behavior and Mental Health, Peking University [6] Yuanpei College, Peking University
[7] Embodied Intelligence Lab, PKU-Wuhan Institute for Artificial Intelligence
[*] Equal contribution ✉ yixin.zhu@pku.edu.cn, zyjiao@buaa.edu.cn https://automoma.pages.dev/

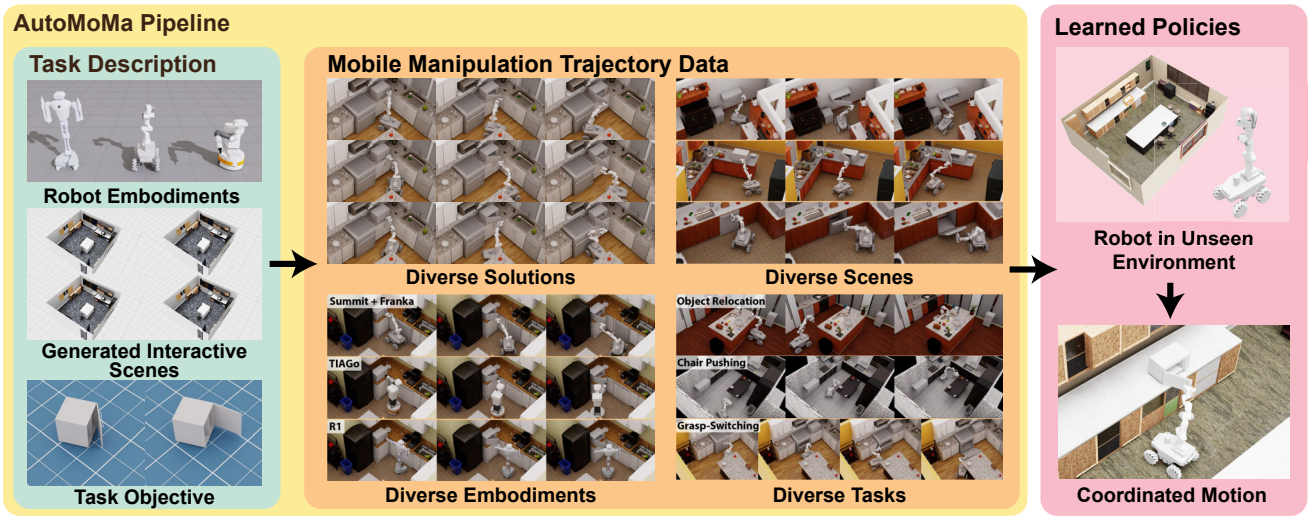


Figure 1. **Overview of the AutoMoMa framework.** Coordinated mobile manipulation demands large-scale, physically valid trajectory data—a bottleneck that existing teleoperation and planning methods cannot overcome at scale. AutoMoMa addresses this by unifying Augmented Kinematic Representation (AKR) modeling, which consolidates base, arm, and object kinematics into a single chain, with GPU-accelerated trajectory optimization. Given diverse robot embodiments, interactive scenes, and task objectives as inputs (left), AutoMoMa efficiently synthesizes over 500k trajectories exhibiting broad diversity across solutions, scenes, embodiments, and complex tasks such as grasp switching (center). This high-quality data enables the training of robust Imitation Learning (IL) policies that generalize coordinated whole-body motion to unseen environments (right).

Abstract

Robots deployed in unstructured environments must coordinate whole-body motion—simultaneously moving a mobile base and arm—to interact with the physical world. This coupled mobility and dexterity yields a state space that grows combinatorially with scene and object diversity, demanding datasets far larger than those sufficient for fixed-base manipulation. Yet existing acquisition methods, including teleoperation [13] and planning [22, 38], are either labor-intensive or computationally prohibitive at scale. The core bottleneck is the lack of a scalable pipeline for generating large-scale, physically valid, coordinated trajectory data across diverse embodiments and environments. Here we introduce AutoMoMa, a GPU-accelerated framework that unifies AKR modeling, which consolidates base, arm, and object kinematics into a single chain, with parallelized trajectory optimization. AutoMoMa achieves 5,000 episodes per GPU-hour (over 80× faster than CPU-based baselines [50]), producing a dataset of over 500k physically valid trajectories spanning 330 scenes, diverse articulated objects, and multiple robot embodiments. Prior datasets were forced to compromise on scale, diversity, or kinematic fidelity [13, 50]; AutoMoMa addresses all three simultaneously. Training downstream IL policies further reveals that even a single articulated-object task requires tens of thousands of demonstrations for State-of-the-Art (SOTA) methods to reach ≈80% success, confirming that data scarcity—not algorithmic limitations—has been the binding constraint. AutoMoMa thus bridges high-performance planning and reliable IL-based control, providing the infrastructure previously missing for coordinated mobile manipulation research. By making large-scale, kinematically valid training data practical, AutoMoMa showcases generalizable whole-body robot policies capable of operating in the diverse, unstructured settings of the real world.

1. Introduction

Whole-body mobile manipulation, which requires coordinated control of the mobile base and the arm, is fundamental for autonomous robots in unstructured environments. Unlike fixed-base systems, mobile manipulators couple base mobility with arm dexterity. This additional base mobility spans the entire room and exponentially expands the search space, making valid kinematic solutions under strict articulation and collision constraints highly sparse [24, 31, 38]. Learning reliable policies for whole-body mobile manipulation therefore necessitates datasets orders of magnitude larger than those sufficient for stationary manipulation.

Yet acquiring such data at scale remains an open challenge. Teleoperation [13] yields high-fidelity demonstrations but is labor-intensive and fundamentally unscalable. Simulation-based Reinforcement Learning (RL) [12, 46] automates data collection but suffers from prohibitively expensive exploration and persistent sim-to-real gaps. Planning-based methods [38] ensure physical validity, yet their CPU-based implementations are computationally prohibitive: the AKR framework [22], which unifies base, arm, and object kinematics into a single representation [21], generates only 60 trajectories per hour [50]. End-to-end learning methods [27, 50] attempt to sidestep these issues by replacing exhaustive search with rapid neural inference, but remain bottlenecked by scarce coordinated whole-body demonstrations. Collectively, these constraints have forced prior datasets to compromise on scale, diversity, or kinematic fidelity [8, 33, 44], leaving generalizable policy learning for whole-body mobile manipulation largely unsolved.

The AKR framework represents a principled foundation for this task, yet its potential has been curtailed by throughput limitations [50]. This bottleneck has fragmented research into narrow-purpose datasets [5, 13], underscoring the need for a framework that bridges precise kinematic modeling with the throughput of modern parallel computing at scale.

We address this with `AutoMoMa`, a scalable framework that integrates AKR modeling with GPU-accelerated motion planning [42]. By batching trajectory optimization and collision checking on the Graphics Processing Unit (GPU), `AutoMoMa` generates physically valid whole-body trajectories at 5,000 episodes per GPU-hour—80× faster than CPU-based baselines—enabling a dataset of over 500k trajectories across 330 realistic scenes, diverse robot morphologies, and articulated objects. The framework further supports complex, multi-step interactions such as grasp switching in confined spaces, ensuring high-fidelity data generation without costly human demonstrations.

Beyond dataset construction, we empirically validate the necessity of this scale. Experiments with downstream policies reveal that even a single articulated-object task requires tens of thousands of demonstrations for current SOTA methods to reach 80% success, confirming that data scarcity—not algorithmic limitations—has been the fundamental binding constraint in learning whole-body mobile manipulation.

In summary, our contributions are: (i) a GPU-accelerated AKR planner that generates physically valid whole-body trajectories at 5,000 episodes per GPU-hour (80× speedup), effectively resolving the acquisition bottleneck; (ii) a comprehensive dataset comprising over 500k trajectories across 330 scenes, covering diverse articulated objects and robot embodiments; and (iii) empirical evidence that SOTA policies (*e.g.*, DP3 [49]) require tens of thousands of demonstrations to achieve high success rates, underscoring the imperative for the scale that `AutoMoMa` provides. Together, these contributions establish `AutoMoMa` as the first framework to bridge high-performance planning and large-scale learning for whole-body mobile manipulation.

2. Related Work

2.1. Motion Planning for Mobile Manipulation

Model-based planning Classical methods for whole-body mobile manipulation often rely on task-specific controllers, such as impedance control for door opening [16, 23, 39], or general base-arm optimization for cluttered scenes [2, 3, 14]. While effective in controlled settings, these methods require extensive hand-tuning and struggle to generalize across diverse object types and environments. The AKR framework [20–22] unifies the base, manipulator, and object into a single kinematic chain, enabling constraint-aware planning in a unified configuration space. However, current implementations rely on CPU-based solvers [50], which are computationally prohibitive for large-scale generation and are typically limited to fixed grasp poses, thereby restricting their utility for diverse dataset creation.

Learning-based planning End-to-end deep RL has been applied to whole-body control in simulation [12, 46], yet it remains highly sample-inefficient and prone to overfitting specific environments [40]. IL offers a more data-efficient alternative [13, 18] but is fundamentally constrained by the availability of high-quality demonstrations. Both paradigms thus share a common dependency: large-scale datasets capturing physically valid whole-body motions, whose absence remains an unresolved bottleneck.

2.2. Data Collection for Mobile Manipulation

Simulated embodied AI platforms Simulators such as Habitat 2.0 [43], AI2-THOR [25], OmniGibson [27], and RoboHive [26] provide photorealistic environments but often prioritize visual fidelity over physical accuracy. Interactions in these platforms are frequently reduced to scripted primitives that bypass the complexities of base-arm and

Table 1. **Comparison of `AutoMoMa` with existing mobile manipulation datasets.** Existing datasets are constrained by their acquisition methods: teleoperation yields high-fidelity but small-scale data, while scripted policies lack whole-body coordination. `AutoMoMa` overcomes these limitations through GPU-accelerated automated planning, simultaneously achieving large scale, broad diversity, and high-fidelity joint-space trajectories—a combination no prior dataset provides. "**Coord.**": presence of whole-body base-arm coordination.

Dataset	Robot	# Episodes	Coord.	# Scenes	Action	Method
RT-1 Robot Action [4]	Google Robot	73,499	Yes	10	End-effector pose	VR teleoperation
NYU VINN [33]	Hello Stretch	435	Yes	3	End-effector pose	Kinesthetic teaching
BC-Z [18]	Google Robot	39,350	Yes	2–3	End-effector pose	VR teleoperation
ETH Agent Affordances [35]	Franka	120	No	50	End-effector pose	Scripted policy
QUT Dexterous Manip. [5]	Franka	200	No	1	End-effector pose	VR teleoperation
CMU Stretch [1, 29]	Hello Stretch	135	No	10	End-effector pose	Scripted Policy
ConqHose [30]	Spot	139	Yes	3	End-effector vel.	Scripted policy
DobbE [36]	Hello Stretch	5,208	Yes	216	End-effector pose	Tool-based teleoperation
Mobile ALOHA [13]	Mobile ALOHA	276	Yes	5	Joint position	Leader-follower teleoperation
TidyBot [44]	TidyBot	24	No	104	Other	Scripted primitives
Ours (`AutoMoMa`)	**Multi-Robot**	**500,000**	**Yes**	**330**	**Joint position**	**Automatic motion planning**

object kinematics. While benchmarks such as ManiSkill-HAB [37] enable policy learning for mobile manipulation, they often restrict training to a narrow set of objects or single-scene layouts, lacking the environmental diversity required for robust generalization.

Teleoperation Human-guided teleoperation captures realistic behaviors but suffers from severe scalability issues. Early systems recorded only end-effector trajectories [45, 48], omitting the joint-space data essential for whole-body motion. Modern platforms such as Mobile ALOHA [13], Behavior Robot Suite [19], and TeleMoMa [9] capture full-body motion but are constrained by operator fatigue and hardware limitations, thereby limiting datasets to the order of thousands. Data augmentation techniques such as MoMaGen [28] aim to scale data by extracting task information from demonstrations and regenerating trajectories, but rely on decoupled planners that generate base and arm motions *separately*, failing to synthesize truly *whole-body* behavior.

Existing mobile manipulation datasets The constraints of these acquisition methods have led to a scarcity of large-scale datasets, as summarized in Tab. 1. Existing resources such as BC-Z [18] contain up to 39,350 episodes but primarily use stationary bases or decoupled base–arm control, while Mobile ALOHA [13] provides high-quality whole-body data but is limited to 276 demonstrations on a single platform. Consequently, current datasets are generally narrow in task coverage, robot diversity, or physical validity [8, 33, 44]. `AutoMoMa` addresses these gaps by leveraging GPU-accelerated planning to automate the generation of over 500k constraint-compliant whole-body trajectories across 330 scenes and multiple robot embodiments.

3. Preliminaries

We outline the AKR-based planning formulation that underlies `AutoMoMa`, covering the AKR construction procedure, the motion planning problem formulation, and the integration of task and physical constraints.

3.1. The Augmented Kinematic Representation

The AKR constructs a serial kinematic chain that integrates the mobile base, the manipulator, and the target object into a unified representation [22], taking three inputs: (i) the robot's kinematic tree, (ii) the object's kinematic tree, and (iii) the transformation between the robot's end-effector and the object's attachable frame (*i.e.*, the grasping pose).

As illustrated in Fig. 2, the robot and object are initially represented as separate kinematic trees (*e.g.*, via Unified Robot Description Format (URDF)). To couple them into a single AKR chain, we attach the object to the robot by inserting a *virtual joint* that encodes the grasp pose between the robot's end-effector and the object; this requires inverting the object's kinematic model so that the attachment link becomes the new kinematic root. Crucially, this inversion extends beyond reversing parent-child relationships: all associated transformations, including branching structures,

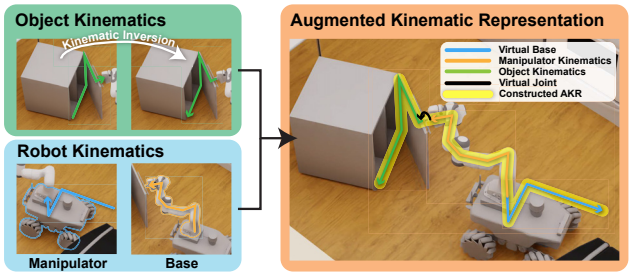


Figure 2. **An example of the AKR construction.** The AKR unifies independent kinematic trees into a single serial chain, enabling joint whole-body optimization of the base, arm, and object. The mobile base's planar motion is modeled via a *virtual base* (blue), while a *virtual joint* (black) couples the manipulator (orange) to the target object (green). For articulated objects, the kinematic tree is *inverted* to reconfigure the kinematic root to the grasp point, forming a continuous chain (highlighted in yellow) rooted at the virtual world frame.

must be rigorously updated, as revolute and prismatic joints typically define motion relative to the child link's frame. The geometry of these branching structures must likewise be preserved during trajectory optimization to ensure collision safety and physical feasibility. Implementation details for kinematic inversion and object-to-robot assembly are provided in Sec. A.1.

To enable joint optimization of mobile robot locomotion and manipulation, we introduce a *virtual base* to model the mobile base's planar motion via two orthogonal prismatic joints and a revolute joint between the world frame and the robot's base, maintaining a strict serial kinematic structure throughout. Fig. 2 illustrates a constructed AKR for the door opening task. The resulting chain is rooted at the world link and terminates at the object's environmental anchor point (*e.g.*, a fixed cabinet base). With the mobile base and manipulator embedded within this serial chain, their states—along with that of the object—are unified within the AKR configuration space, within which task goals and kinematic constraints are subsequently enforced during trajectory optimization.

3.2. AKR-Based Mobile Manipulation Planning

We formulate the whole-body mobile manipulation planning problem as finding a collision-free trajectory within the unified AKR configuration space that satisfies both kinematic and task-specific constraints. Formally, the AKR state is defined as:

$$\boldsymbol{x} = \left[\boldsymbol{q}_B^\mathsf{T}, \boldsymbol{q}_M^\mathsf{T}, \boldsymbol{q}_O^\mathsf{T}\right]^\mathsf{T} \in \mathcal{X}_{\text{free}}, \tag{1}$$

where $\boldsymbol{q}_B \in \mathbb{R}^3$ denotes the mobile base pose, $\boldsymbol{q}_M \in \mathbb{R}^n$ represents the manipulator joint configuration (n is the arm Degree of Freedom (DoF)), and $\boldsymbol{q}_O \in \mathbb{R}^m$ is the joint state of the articulated object (m is the object DoF, with $m = 0$ for rigid objects). The planning objective is to generate a valid trajectory of length T: $\boldsymbol{x}_{1:T} = \langle \boldsymbol{x}_{[1]}, \ldots, \boldsymbol{x}_{[T]} \rangle \subset \mathcal{X}_{\text{free}}$, where $\mathcal{X}_{\text{free}}$ represents the collision-free configuration space. Following Jiao et al. [22], we enforce the following constraints during trajectory optimization:

$$h_{\text{chain}}(\boldsymbol{x}_{[t]}) = 0, \qquad \forall t = 1, \ldots, T, \tag{2}$$

$$\|f_{\text{task}}(\boldsymbol{x}_{[T]}) - \boldsymbol{g}_{\text{goal}}\|_2^2 \leqslant \xi_{\text{goal}}, \tag{3}$$

$$\boldsymbol{x}_{\min} \leqslant \boldsymbol{x}_{[t]} \leqslant \boldsymbol{x}_{\max}, \qquad \forall t = 1, \ldots, T, \tag{4}$$

$$\|\Delta \boldsymbol{x}_{[t]}\|_\infty \leqslant \Delta \boldsymbol{x}_{\max}, \qquad \forall t = 1, \ldots, T-1, \tag{5}$$

$$\|\Delta \dot{\boldsymbol{x}}_{[t]}\|_\infty \leqslant \Delta \dot{\boldsymbol{x}}_{\max}, \qquad \forall t = 2, \ldots, T-1. \tag{6}$$

Eq. (2) enforces kinematic constraints imposed by the object's attachment to the environment, such as a revolute door's hinge connection or a sliding chair's planar constraint. Eq. (3) ensures task completion by bounding the terminal state within a tolerance ξ_{goal} of the target goal $\boldsymbol{g}_{\text{goal}}$, as mapped by the task function $f_{\text{task}} : \mathcal{X} \to \mathcal{G}$. Eq. (4)–Eq. (6) impose physical limits on joint positions, velocities, and accelerations. Collision avoidance is handled implicitly via integrated self- and environment-collision checks within the underlying motion planner.

4. The AutoMoMa Data Generation Pipeline

AutoMoMa synthesizes large-scale, physically valid whole-body trajectory data through four integrated stages—task specification, problem instantiation, trajectory generation, and rendering—as illustrated in Fig. 3. We describe each stage in turn.

4.1. Task Specification

The pipeline accepts a task specification triplet $(\mathcal{S}, \mathcal{O}, \mathcal{R})$ that defines the semantic and geometric context of each mobile manipulation task.

Household scene layouts Each scene $\mathcal{S}$ encapsulates the geometric, visual, and semantic properties of structural elements such as walls, floors, and static appliances. All entities are anchored to a central world frame and associated with high-resolution visual and collision meshes. To maximize environmental diversity, our layouts are sourced via two complementary strategies: (i) procedural generation of interactive scenes with articulated appliances, and (ii) augmentation of existing scene datasets by substituting static appliances with functionally equivalent articulated counterparts. Details of scene construction are provided in Sec. B.1.

Interactive objects The object set $\mathcal{O} = \mathcal{O}_{\text{rigid}} \cup \mathcal{O}_{\text{art}}$ includes both rigid and articulated entities. Rigid objects $o \in \mathcal{O}_{\text{rigid}}$ are characterized by watertight meshes and static grasp poses, whereas articulated objects $o \in \mathcal{O}_{\text{art}}$ require comprehensive URDF descriptions specifying kinematic chains, joint limits, and inertial parameters. Crucially, the framework accounts for state-dependent grasp poses in articulated objects (*e.g.*, handles moving during interaction); as detailed in Sec. 3.1, we re-root the object at the grasp point by inverting its kinematic tree, allowing the object to be attached to the end-effector and optimized jointly with the robot in a unified AKR configuration space.

Robot embodiments A robot embodiment $\mathcal{R}$ comprises a virtual mobile base and a manipulator, both defined via URDF. To facilitate high-throughput GPU-accelerated planning, each embodiment is augmented with a spherical approximation of its collision geometry, a self-collision mask to prune permanent contacts, and a joint-weight vector $\boldsymbol{w} \in \mathbb{R}^{n+m+3}$ to modulate the optimization cost. The framework is embodiment-agnostic, as demonstrated across diverse platforms including a Franka arm on the Summit base, the R1 robot, and the TIAGo mobile manipulator.

4.2. Problem Instantiation

AutoMoMa instantiates the planning problem by transforming raw scene geometries into structured representa-

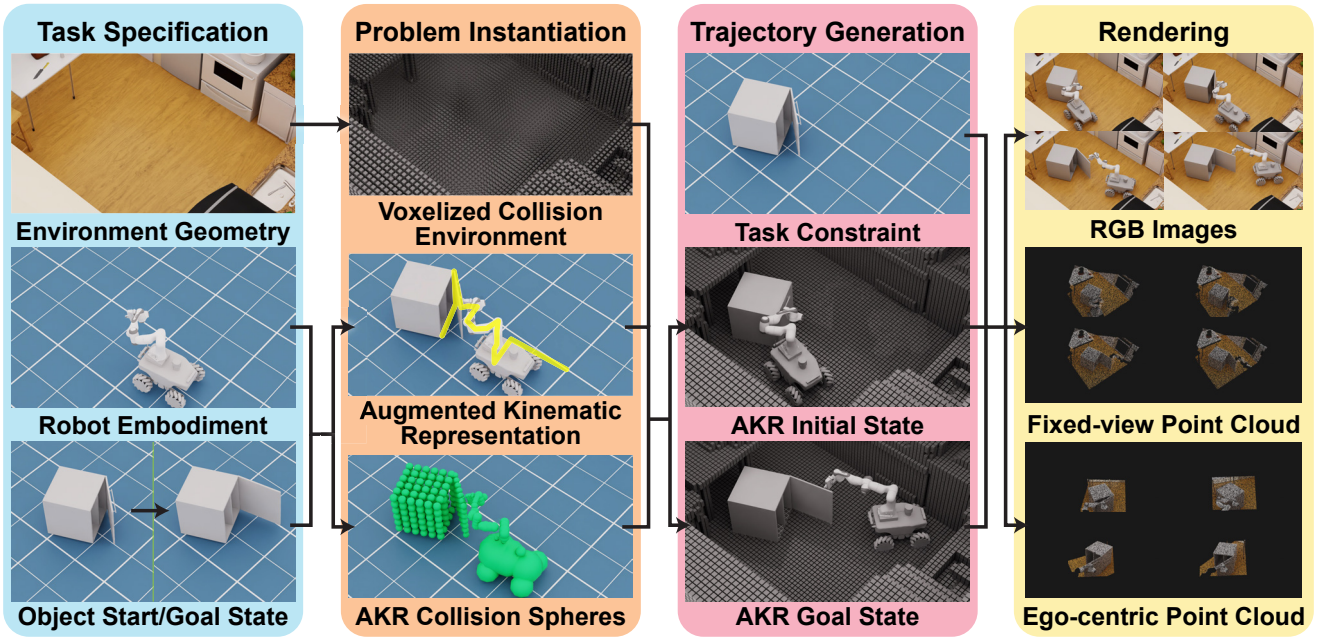


Figure 3. **The `AutoMoMa` data generation pipeline.** Starting from a task specification triplet $(\mathcal{S}, \mathcal{O}, \mathcal{R})$ (left), `AutoMoMa` proceeds through four stages: (i) **Task Specification** defines the environmental, robotic, and object context; (ii) **Problem Instantiation** transforms raw scene assets into planning-ready primitives via ESDF construction and AKR assembly with spherical collision approximations; (iii) **Trajectory Generation** solves for optimal AKR states under task-specific constraints to produce physically valid whole-body motions; and (iv) **Rendering** in NVIDIA Isaac Sim produces synchronized RGB-D sequences and point clouds. The resulting trajectories span diverse scenes, objects, and robot embodiments (right).

tions that bridge environmental context with unified robot-object kinematics.

Environment collision models To accelerate environmental queries, each scene is converted into an Euclidean Signed-distance Field (ESDF). Efficiency is further enhanced by restricting planning queries to an axis-aligned bounding box defined by the object's start and goal states, thereby focusing computation on the local workspace and minimizing overhead.

AKR construction The AKR construction integrates the processed object model with the robot's kinematics into a single kinematic chain. To account for varying scene contexts, objects are first resized to align with the environment, with link components merged into single meshes for uniform scaling. Joint origins are then rigorously recalculated to compensate for the resulting spatial shifts and maintain kinematic integrity. This preprocessed model is coupled to the robot's end-effector via a virtual joint at the grasp pose, yielding a unified model $\mathcal{K}_{\text{akr}}$ in which the object effectively becomes a kinematic extension of the robot.

Collision processing For high-throughput GPU-accelerated planning, link geometries are approximated using fitted spheres. Meshes are downscaled before sphere fitting to prevent volume overestimation and ensure conservative collision avoidance. When voxelization induces translational offsets, sphere cloud centroids are realigned with the original meshes to preserve geometric fidelity. Negligible collision pairs, such as permanently contacting adjacent links, are additionally masked to reduce computational overhead. To manage collision artifacts across task phases, we employ a dynamic strategy. In the *approach* phase, environmental voxels intersecting the object are temporarily cleared and replaced with the high-resolution mesh, preventing discretization errors from blocking valid object grasp poses. In the *manipulation* phase, the object transitions to a link of the AKR and its static environmental mesh is removed; only voxels lying strictly outside the object's current volume remain active, eliminating false-positive collisions with the object's initial state. Sec. A.2 details the collision-sphere fitting and alignment procedure.

4.3. Trajectory Generation

`AutoMoMa` synthesizes whole-body trajectories by formulating and solving a constrained optimization problem within the unified AKR configuration space, enabling precise goal specification while simultaneously enforcing task-specific constraints across the mobile base, manipulator, and object states.

Task objectives and goals The planning objective $\mathcal{J}$ minimizes both total travel distance and trajectory non-smoothness. Let $\boldsymbol{x}_{1:T}$ denote the trajectory over T time

steps:

$$\mathcal{J}(\boldsymbol{x}_{1:T}) = \sum_{t=1}^{T-1} \left\| \boldsymbol{w}_v \, \Delta \boldsymbol{x}_{[t]} \right\|_2^2 + \sum_{t=2}^{T-1} \left\| \boldsymbol{w}_a \, \Delta \dot{\boldsymbol{x}}_{[t]} \right\|_2^2, \quad (7)$$

$$\boldsymbol{x}^{\star}_{1:T} = \arg\min_{\boldsymbol{x}_{1:T}} \mathcal{J}(\boldsymbol{x}_{1:T}), \quad (8)$$

where diagonal weight matrices $\mathbf{W}_v$ and $\mathbf{W}_a$ modulate coordination strategies, such as prioritizing base stability during interaction. Task goals are defined according to object type: target $SE(3)$ poses for rigid object relocation, or specific joint configurations (*e.g.*, door opening angles) for articulated objects.

Task constraints Trajectory constraints are derived from the semantic and physical relationships between the object and the scene. Rigid objects are modeled as free-floating, while heavy entities such as chairs are restricted to $SE(2)$ planar motion. For stationary articulated objects, a strict pose constraint is enforced on the AKR end-effector to penalize deviations from the object's base link, effectively modeling its physical attachment to the environment.

Optimization problem formulation Valid AKR start and goal configurations are computed by solving Inverse Kinematics (IK) for the respective object states. To manage computational overhead while ensuring diverse configuration-space coverage, similar IK solutions are clustered in joint space, retaining a compact set of representative candidate configurations. For complex tasks where kinematic limits or collisions prevent a continuous grasp (*e.g.*, opening a dishwasher in a confined space), `AutoMoMa` employs a multi-stage strategy, sampling an intermediate state ϕ_{mid} to connect two trajectory segments—$[\phi_0 \to \phi_{\text{mid}}]$ and $[\phi_{\text{mid}} \to \phi_T]$—via a collision-free re-grasp action. Qualitative examples are provided in Sec. E.4.

Trajectory post-processing Optimized trajectories are filtered to remove solutions that violate task constraints and ensure kinematic consistency. Each waypoint $\boldsymbol{x}_{[t]}$ is validated against the required constraints. For stationary articulated objects, we evaluate translational deviation d and rotational deviation θ for the object-world attachment:

$$\begin{aligned} d &= \|p(\boldsymbol{x}_{[t]}) - p(\boldsymbol{x}_{\text{ref}})\|_2, \\ \theta &= \arccos(2\langle r(\boldsymbol{x}_{[t]}), r(\boldsymbol{x}_{\text{ref}})\rangle^2 - 1), \end{aligned} \quad (9)$$

where $p(\cdot)$ and $r(\cdot)$ denote the positional and rotational components of the AKR Forward Kinematics (FK). For planar constraints, vertical displacement d_z and orientation deviation θ_{planar} are additionally bounded:

$$\begin{aligned} d_z &= |p_z(\boldsymbol{x}_{[t]}) - p_z(\boldsymbol{x}_{\text{ref}})|, \\ \theta_{\text{planar}} &= \|\psi(\boldsymbol{x}_{[t]}) - \psi(\boldsymbol{x}_{\text{ref}})\|_2, \end{aligned} \quad (10)$$

where $p_z(\cdot)$ is the z-axis translation and $\psi(\cdot)$ represents roll and pitch angles. Trajectories violating these thresholds are discarded, ensuring the final dataset contains only stable, physically plausible whole-body motions.

4.4. Rendering

The final pipeline stage uses NVIDIA Isaac Sim to synthesize high-fidelity multi-modal observations from validated trajectories. Synchronized egocentric and fixed-viewpoint Red, Green, Blue - Depth (RGB-D) cameras are configured on both the robot platform and within the environment to ensure multi-perspective coverage. At each waypoint $\boldsymbol{x}_{[t]}$, RGB and depth images are rendered and projected into 3D point clouds within the simulation world coordinate frame, pairing every joint-space configuration with its corresponding geometric and visual context.

The rendering framework is designed for extensibility: camera placements are fully customizable, and saved trajectories can be replayed under varying lighting conditions, camera configurations, or sensor modalities. The resulting dataset provides a robust foundation for diverse downstream tasks, including IL [11, 15], visual servoing [17, 41], and affordance detection [7, 10].

5. Experiments

We evaluate `AutoMoMa` along two dimensions: (i) characterizing the diversity and generation efficiency of the synthesized dataset; and (ii) empirically investigating how data scale and diversity affect policy learning for whole-body mobile manipulation.

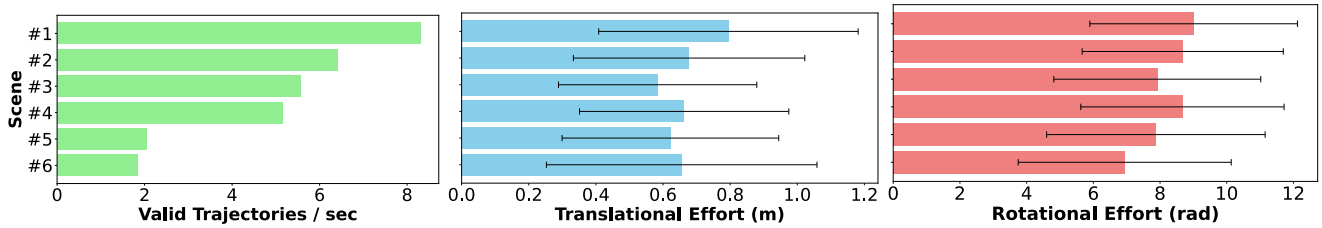
5.1. Dataset Statistics and Diversity Analysis

`AutoMoMa` integrates existing virtual household environments [25, 34] populated with articulated objects sourced from PartNet-Mobility [47]. Leveraging three distinct robot platforms—Summit Franka, TIAGo, and R1—we generated over 500k physically valid trajectories. Each trajectory comprises 30 joint-space waypoints accompanied by synchronized multi-modal observations, including RGB-D images and point clouds (4,096 points per frame) rendered at 120 frames per trajectory.

Grasp and configuration diversity To ensure broad coverage of the configuration space, each object is paired with ∼20 AO-Grasp [32] annotations. We compute approximately 30 IK solutions per grasp and cluster them in joint space to retain a diverse set of representative start states. This sampling strategy (Fig. 5) ensures that trajectories span a broad distribution of feasible robot base placements.

Pipeline performance benchmarks We benchmark the trajectory generation pipeline across six representative household scenes (Fig. 4a) with varying spatial constraints. Less cluttered layouts yield higher throughput, whereas confined spaces increase collision-checking overhead and reduce feasible IK counts (Fig. 4b). To further characterize planner behavior, we measure the average translational motion of the base (Fig. 4c) and cumulative arm rotation (Fig. 4d); these metrics reflect the planner's ability to syn-

(a) Scene layouts (#1–#6), ordered by increasing spatial confinement.



(b) Generation throughput. (c) Ave. translational effort of the base. (d) Ave. rotational effort of the manipulator.

Figure 4. **Trajectory generation performance across six representative household scenes.** (a) Test scenes with increasing spatial confinement. (b) Generation throughput (valid trajectories per second) decreases as scene clutter increases collision-checking overhead. (c) Average translational effort of the mobile base per trajectory (error bars: standard deviation). (d) Average rotational effort of the manipulator, reflecting compensatory whole-body motion in constrained environments.

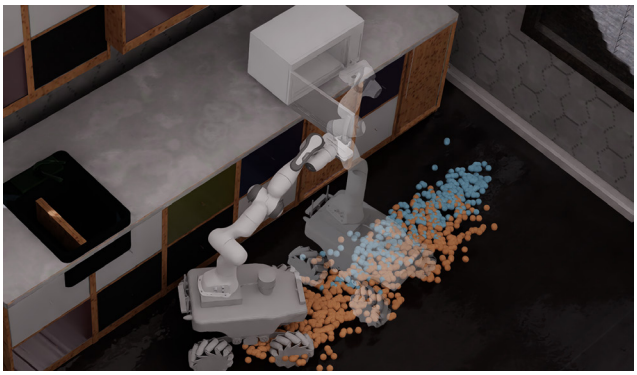


Figure 5. **Distribution of trajectory base positions.** Blue and orange spheres denote start and goal base placements, respectively, illustrating the broad spatial coverage achieved by the IK clustering strategy.

thesize compensatory whole-body motions in diverse and restrictive environments.

5.2. Policy Learning Setup

We evaluate the utility of the synthesized trajectories by training IL policies for whole-body coordination across three representative architectures.

Physical simulation All experiments are conducted in Isaac Sim, utilizing its GPU-accelerated PhysX engine to simulate complex articulated interactions with high-fidelity physical feedback and synchronized sensor rendering.

Agent and observation space We employ the Summit Franka mobile manipulator as our primary agent. The policy architecture is DP3 [49], a SOTA diffusion-based method used to benchmark data scaling laws. To demonstrate that the benefits of `AutoMoMa` are model-agnostic, we additionally evaluate DP (RGB-based Diffusion Policy) [6] and ACT (Transformer Policy) [13]. The observation space consists of fused point clouds (4,096 points) aggregated from egocentric and fixed RGB-D cameras, together with proprioceptive states (joint positions and base pose). All visual inputs are rendered at 320×240 resolution.

Training and evaluation Models are trained for 300 epochs with a batch size of 256 using the AdamW optimizer ($lr = 1 \times 10^{-4}$). Policies are evaluated on a microwave door opening task; a trial is successful if the door reaches the target angle within 300 steps, with success rates averaged over 50 randomized trials per setting. Complete hyperparameters are reported in Secs. B.2 and C.

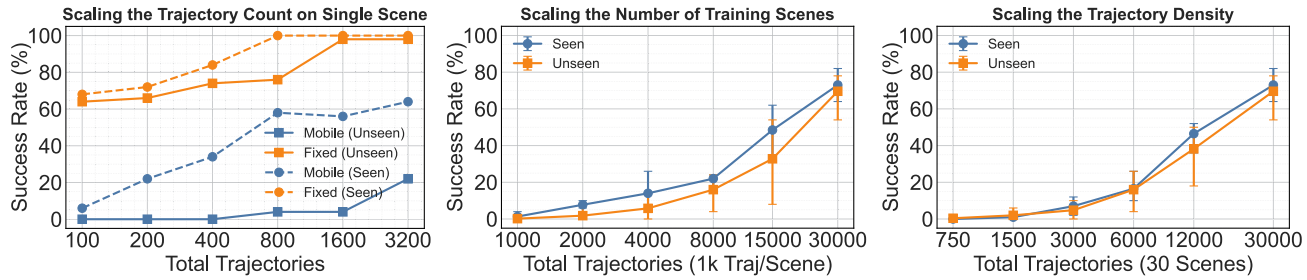
5.3. Result Analysis

We analyze policy performance across six dimensions to characterize how data scale and diversity affect whole-body mobile manipulation learning.

Configuration space complexity We first investigate the complexity gap between fixed-base and mobile manipulation (Fig. 6a). The fixed-base robot achieves 100% success with fewer than 800 trajectories, as its lower-dimensional configuration space is easily covered. In contrast, the mobile base policy saturates at approximately 70% success on *seen* configurations even with 3,200 trajectories. This performance gap stems from the complex 10-DoF base-arm coupling, exponentially expanding the search space and requiring massive data for robust coordination.

Local generalization Within a single-scene context, we compare performance on *seen* and *unseen* IK states (Fig. 6a). Despite strong performance on seen start configurations, the policy degrades significantly on novel IK starts within the same workspace, indicating that high trajectory density in a single environment promotes manifold memorization rather than scene understanding. This underscores the need for broader diversity in the planning context.

Environmental diversity scaling Scaling from 1 to 30 scenes (Fig. 6b) shows that policies trained on limited



(a) Fixed *vs*. Mobile Base on single scene. (b) Scene count scaling with 1k traj/scene. (c) Trajectory density scaling on 30 scenes.

Figure 6. **Data scaling experiments.** (a) In a single scene, the mobile base policy requires substantially more data than the fixed-base counterpart, with a persistent seen/unseen gap indicating manifold memorization. (b) Increasing scene diversity from 1 to 30 steadily improves generalization to unseen environments. (c) With 30 scenes, higher per-scene trajectory density further refines execution precision, enabling consistent generalization across seen and unseen scenes.

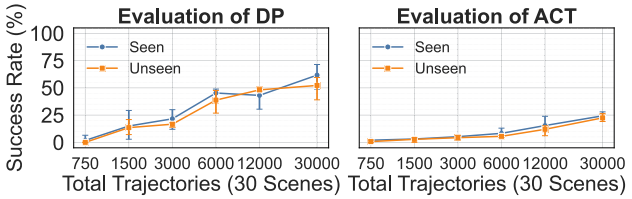


Figure 7. **Architectural generalization of `AutoMoMa`.** When evaluated across the same 30-scene setup as DP3 [49], both DP [6] and ACT [13] exhibit consistent performance gains with increasing trajectory density, demonstrating `AutoMoMa`'s compatibility with diverse whole-body IL architectures.

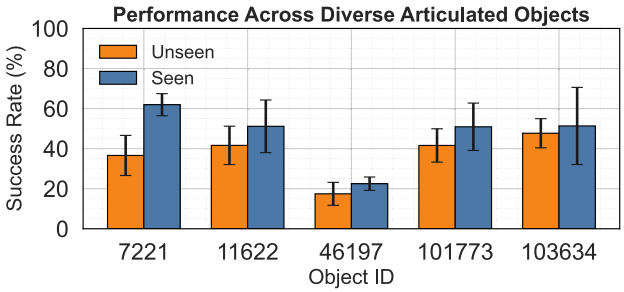


Figure 8. **Per-object success rates at 100k trajectories.** Success rates of the DP3 policy evaluated on five representative SAPIEN [47] objects. The bar plot compares performance under unseen (orange) and seen (blue) environments.

scenes fail to generalize to novel layouts due to overfitting to specific clutter and geometry. As scene diversity increases, success rates on *unseen* environments improve steadily, demonstrating that geometric variety from the integrated iTHOR and Infinigen environments is the primary driver for learning transferable whole-body strategies.

Trajectory density in multi-scene scaling Fixing the environment count at 30 and varying per-scene trajectory density (Fig. 6c), we find that increasing trajectory density yields comparable improvements to expanding scene diversity. This higher density enables the model to capture a richer distribution of approach angles and motion variations, allowing the policy to generalize consistently across both seen and unseen scenes ($\sim$75% success).

Architectural generalization We also evaluate `AutoMoMa` on DP [6] and ACT [13] (Fig. 7). Both architectures exhibit consistent gains as trajectory density increases, though DP3 remains superior given its 3D modalities. This demonstrates the broad applicability of `AutoMoMa` across different whole-body IL models. Further discussion is in Sec. E.1.

Performance stability across objects We evaluate five representative SAPIEN [47] objects using the DP3 policy trained at the 100k trajectory scale (Fig. 8). The policy achieves over 50% success on seen configurations for the majority of the evaluated objects (IDs 7221, 11622, 101773, and 103634), confirming robust scaling benefits across diverse kinematic constraints. Variance in specific objects (*e.g.*, ID 46197) stems from articulation constraints limiting the robot workspace rather than data scarcity.

Qualitative inference rollouts, representative failure modes, and experiments on rigid-object picking are provided in Secs. D.2 and E.3.

6. Limitations and Conclusion

`AutoMoMa` provides a scalable framework that generates over 500k physically valid whole-body trajectories across diverse scenes, objects, and embodiments. While we validate these trajectories in the real world on a UR5-Ridgeback platform (see Sec. D.3), several limitations remain. The current pipeline relies on known scene geometries and kinematics, and does not support dynamic human-robot interaction or deformable objects. Additionally, the sphere-based collision approximations required for GPU acceleration can occasionally introduce geometric inaccuracies that cause execution failures (see Sec. D.1). Future work will integrate learning-based generation methods and develop community-driven tools that facilitate seamless extension to new robots and environments, further broadening `AutoMoMa`'s utility as a foundation for embodied AI.

Acknowledgment

This work is supported in part by the National Science and Technology Innovation 2030 Major Program (2025ZD0219400), the National Natural Science Foundation of China (62376009 to Y.Z., and 52305007 to Z.J.), the PKU-BingJi Joint Laboratory for Artificial Intelligence, the Wuhan Major Scientific and Technological Special Program (2025060902020304), the Hubei Embodied Intelligence Foundation Model Research and Development Program, and the National Comprehensive Experimental Base for Governance of Intelligent Society, Wuhan East Lake High-Tech Development Zone.

References

[1] Shikhar Bahl, Russell Mendonca, Lili Chen, Unnat Jain, and Deepak Pathak. Affordances from human videos as a versatile representation for robotics. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[2] Dmitry Berenson, James Kuffner, and Howie Choset. An optimization approach to planning for mobile manipulation. In *International Conference on Robotics and Automation (ICRA)*, 2008. 2

[3] Daniel M Bodily, Thomas F Allen, and Marc D Killpack. Motion planning for mobile robots using inverse kinematics branching. In *International Conference on Robotics and Automation (ICRA)*, 2017. 2

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. In *Robotics: Science and Systems (RSS)*, 2023. 3

[5] Federico Ceola, Lorenzo Natale, Niko Sünderhauf, and Krishan Rana. Lhmanip: A dataset for long-horizon language-grounded manipulation tasks in cluttered tabletop environments. *arXiv preprint arXiv:2312.12036*, 2023. 2, 3

[6] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *International Journal of Robotics Research (IJRR)*, 44(10-11):1684–1704, 2025. 7, 8, A2, A7

[7] Fu-Jen Chu, Ruinian Xu, Landan Seguin, and Patricio A Vela. Toward affordance detection and ranking on novel objects for real-world robotic manipulation. *IEEE Robotics and Automation Letters (RA-L)*, 4(4):4070–4077, 2019. 6

[8] Wenbo Cui, Chengyang Zhao, Songlin Wei, Jiazhao Zhang, Haoran Geng, Yaran Chen, and He Wang. Gapartmanip: a large-scale dataset for generalizable and actionable part manipulation with material-agnostic articulated objects. In *International Conference on Robotics and Automation (ICRA)*, 2025. 2, 3

[9] Shivin Dass, Wensi Ai, Yuqian Jiang, Samik Singh, Jiaheng Hu, Ruohan Zhang, Peter Stone, Ben Abbatematteo, and Roberto Martín-Martín. Telemoma: A modular and versatile teleoperation system for mobile manipulation. *arXiv preprint arXiv:2403.07869*, 2024. 3

[10] Thanh-Toan Do, Anh Nguyen, and Ian Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *International Conference on Robotics and Automation (ICRA)*, 2018. 6

[11] Bin Fang, Shidong Jia, Di Guo, Muhua Xu, Shuhuan Wen, and Fuchun Sun. Survey of imitation learning for robotic manipulation. *International Journal of Intelligent Robotics and Applications*, 3(4):362–369, 2019. 6

[12] Zipeng Fu, Xuxin Cheng, and Deepak Pathak. Deep whole-body control: learning a unified policy for manipulation and locomotion. In *Conference on Robot Learning (CoRL)*, 2023. 2

[13] Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation using low-cost whole-body teleoperation. In *Conference on Robot Learning (CoRL)*, 2024. 1, 2, 3, 7, 8, A4, A7

[14] Kalin Gochev, Alla Safonova, and Maxim Likhachev. Planning with adaptive dimensionality for mobile manipulation. In *International Conference on Robotics and Automation (ICRA)*, 2012. 2

[15] Jiang Hua, Liangcai Zeng, Gongfa Li, and Zhaojie Ju. Learning for a robot: Deep reinforcement learning, imitation learning, transfer learning. *Sensors*, 21(4):1278, 2021. 6

[16] Advait Jain and Charles C Kemp. Pulling open doors and drawers: Coordinating an omni-directional base and a compliant arm with equilibrium point control. In *International Conference on Robotics and Automation (ICRA)*, 2010. 2

[17] Farrokh Janabi-Sharifi, Lingfeng Deng, and William J Wilson. Comparison of basic visual servoing methods. *Transactions on Mechatronics (TMECH)*, 16(5):967–983, 2010. 6

[18] Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning (CoRL)*, 2022. 2, 3

[19] Yunfan Jiang, Ruohan Zhang, Josiah Wong, Chen Wang, Yanjie Ze, Hang Yin, Cem Gokmen, Shuran Song, Jiajun Wu, and Li Fei-Fei. Behavior robot suite: Streamlining real-world whole-body manipulation for everyday household activities. In *Conference on Robot Learning (CoRL)*, 2025. 3

[20] Ziyuan Jiao, Zeyu Zhang, Xin Jiang, David Han, Song-Chun Zhu, Yixin Zhu, and Hangxin Liu. Consolidating kinematic models to promote coordinated mobile manipulations. In *International Conference on Intelligent Robots and Systems (IROS)*, 2021. 2

[21] Ziyuan Jiao, Zeyu Zhang, Weiqi Wang, David Han, Song-Chun Zhu, Yixin Zhu, and Hangxin Liu. Efficient task planning for mobile manipulation: a virtual kinematic chain perspective. In *International Conference on Intelligent Robots and Systems (IROS)*, 2021. 2

[22] Ziyuan Jiao, Yida Niu, Zeyu Zhang, Yangyang Wu, Yao Su, Yixin Zhu, Hangxin Liu, and Song-Chun Zhu. Integration of robot and scene kinematics for sequential mobile manipulation planning. *Transactions on Robotics (T-RO)*, 2025. 1, 2, 3, 4

[23] Yiannis Karayiannidis, Christian Smith, Francisco Eli Vina Barrientos, Petter Ögren, and Danica Kragic. An adaptive

control approach for opening doors and drawers under uncertainties. *Transactions on Robotics (T-RO)*, 32(1):161–175, 2016. 2

[24] Oussama Khatib. Mobile manipulation: The robotic assistant. *Robotics and Autonomous Systems*, 26(2-3):175–183, 1999. 2

[25] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, et al. Ai2-thor: An interactive 3d environment for visual ai. *arXiv preprint arXiv:1712.05474*, 2017. 2, 6, A2

[26] Vikash Kumar, Rutav Shah, Gaoyue Zhou, Vincent Moens, Vittorio Caggiano, Abhishek Gupta, and Aravind Rajeswaran. Robohive: A unified framework for robot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[27] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning (CoRL)*, 2023. 2

[28] Chengshu Li, Mengdi Xu, Arpit Bahety, Hang Yin, Yunfan Jiang, Huang Huang, Josiah Wong, Sujay Garlanka, Cem Gokmen, Ruohan Zhang, et al. Momagen: Generating demonstrations under soft and hard constraints for multi-step bimanual mobile manipulation. *arXiv preprint arXiv:2510.18316*, 2025. 3, A6

[29] Russell Mendonca, Shikhar Bahl, and Deepak Pathak. Structured world models from human videos. In *Robotics: Science and Systems (RSS)*, 2023. 3

[30] Peter Mitrano and Dmitry Berenson. Conq hose manipulation dataset, v1.15.0, 2024. 3

[31] Mayank Mittal, David Hoeller, Farbod Farshidian, Marco Hutter, and Animesh Garg. Articulated object interaction in unknown scenes with whole-body mobile manipulation. In *International Conference on Intelligent Robots and Systems (IROS)*, 2022. 2

[32] Carlota Parés Morlans, Claire Chen, Yijia Weng, Michelle Yi, Yuying Huang, Nick Heppert, Linqi Zhou, Leonidas Guibas, and Jeannette Bohg. Ao-grasp: Articulated object grasp generation. In *International Conference on Intelligent Robots and Systems (IROS)*, 2024. 6

[33] Jyothish Pari, Nur Muhammad Mahi Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The surprising effectiveness of representation learning for visual imitation. In *Robotics: Science and Systems (RSS)*, 2022. 2, 3

[34] Alexander Raistrick, Lingjie Mei, Karhan Kayan, David Yan, Yiming Zuo, Beining Han, Hongyu Wen, Meenal Parakh, Stamatis Alexandropoulos, Lahav Lipson, et al. Infinigen indoors: Photorealistic indoor scenes using procedural generation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 6, A2

[35] Giulio Schiavi, Paula Wulkop, Giuseppe Rizzi, Lionel Ott, Roland Siegwart, and Jen Jen Chung. Learning agent-aware affordances for closed-loop interaction with articulated objects. In *International Conference on Robotics and Automation (ICRA)*, 2023. 3

[36] Nur Muhammad Mahi Shafiullah, Anant Rai, Haritheja Etukuru, Yiqian Liu, Ishan Misra, Soumith Chintala, and Lerrel Pinto. On bringing robots home. *arXiv preprint arXiv:2311.16098*, 2023. 3

[37] Arth Shukla, Stone Tao, and Hao Su. Maniskill-hab: A benchmark for low-level manipulation in home rearrangement tasks. In *International Conference on Learning Representations (ICLR)*, 2025. 3

[38] Jean-Pierre Sleiman, Farbod Farshidian, and Marco Hutter. Versatile multicontact planning and control for legged loco-manipulation. *Science Robotics*, 8(81):eadg5014, 2023. 1, 2

[39] Marvin Stuede, Kathrin Nuelle, Svenja Tappe, and Tobias Ortmaier. Door opening and traversal with an industrial cartesian impedance controlled mobile robot. In *International Conference on Robotics and Automation (ICRA)*, 2019. 2

[40] Charles Sun, Jedrzej Orbik, Coline Manon Devin, Brian H Yang, Abhishek Gupta, Glen Berseth, and Sergey Levine. Fully autonomous real-world reinforcement learning with applications to mobile manipulation. In *Conference on Robot Learning (CoRL)*, 2022. 2

[41] Xiaoying Sun, Xiaojun Zhu, Pengyuan Wang, and Hua Chen. A review of robot control with visual servoing. In *2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, 2018. 6

[42] Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, et al. Curobo: Parallelized collision-free robot motion generation. In *International Conference on Robotics and Automation (ICRA)*, 2023. 2, A2

[43] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: training home assistants to rearrange their habitat. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2

[44] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. Tidybot: Personalized robot assistance with large language models. *Autonomous Robots*, 47(8):1087–1102, 2023. 2, 3

[45] Yuqiang Wu, Pietro Balatti, Marta Lorenzini, Fei Zhao, Wansoo Kim, and Arash Ajoudani. A teleoperation interface for loco-manipulation control of mobile collaborative robotic assistant. *IEEE Robotics and Automation Letters (RA-L)*, 4(4): 3593–3600, 2019. 3

[46] Fei Xia, Chengshu Li, Roberto Martín-Martín, Or Litany, Alexander Toshev, and Silvio Savarese. Relmogen: Integrating motion generation in reinforcement learning for mobile manipulation. In *International Conference on Robotics and Automation (ICRA)*, 2021. 2

[47] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive

environment. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 6, 8, A2

[48] Taozheng Yang, Ya Jing, Hongtao Wu, Jiafeng Xu, Kuankuan Sima, Guangzeng Chen, Qie Sima, and Tao Kong. Moma-force: Visual-force imitation for real-world mobile manipulation. In *International Conference on Intelligent Robots and Systems (IROS)*, 2023. 3

[49] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Robotics: Science and Systems (RSS)*, 2024. 2, 7, 8, A2, A6, A10

[50] Zeyu Zhang, Sixu Yan, Muzhi Han, Zaijin Wang, Xinggang Wang, Song-Chun Zhu, and Hangxin Liu. M3bench: Benchmarking whole-body motion generation for mobile manipulation in 3d scenes. *IEEE Robotics and Automation Letters (RA-L)*, 2025. 1, 2