# Physics-Driven Data Generation for Contact-Rich Manipulation via Trajectory Optimization

Lujie Yang[1,2], H.J. Terry Suh*[1], Tong Zhao*[2], Bernhard Paus Græsdal[1], Tarik Kelestemur[2], Jiuguang Wang[2], Tao Pang[2], and Russ Tedrake[1]

***Abstract*—We present a low-cost data generation pipeline that integrates physics-based simulation, human demonstrations, and model-based planning to efficiently generate large-scale, high-quality datasets for contact-rich robotic manipulation tasks. Starting with a small number of embodiment-flexible human demonstrations collected in a virtual reality simulation environment, the pipeline refines these demonstrations using optimization-based kinematic retargeting and trajectory optimization to adapt them across various robot embodiments and physical parameters. This process yields a diverse, physically consistent, contact-rich dataset that enables cross-embodiment data transfer, and offers the potential to reuse legacy datasets collected under different hardware configurations or physical parameters. We validate the pipeline's effectiveness by training diffusion policies from the generated datasets for challenging long-horizon contact-rich manipulation tasks across multiple robot embodiments, including a floating Allegro hand and bimanual robot arms. The trained policies are deployed zero-shot on hardware for bimanual iiwa arms, achieving high success rates with minimal human input. Project website: https://lujieyang.github.io/physicsgen/.**

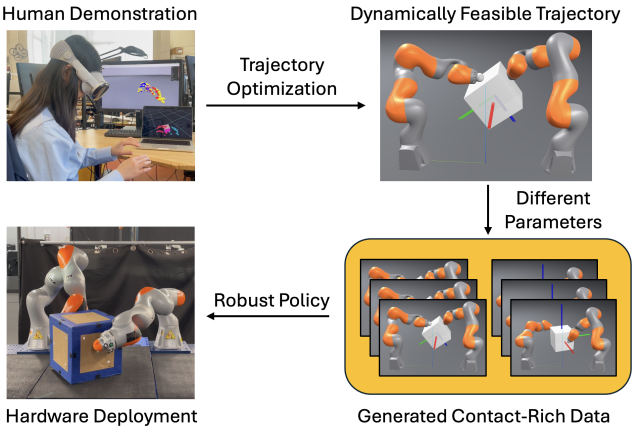


Fig. 1: **Physics-driven data generation overview.** Leveraging trajectory optimization, our framework automatically generates thousands of dynamically feasible contact-rich trajectories across a range of embodiments and physical parameters from only 24 human demonstrations. The policy trained with imitation learning from the generated dataset is more robust and performant.

## I. Introduction

The emergence of foundation models has transformed fields such as natural language processing and computer vision, where models trained on massive, internet-scale datasets demonstrate remarkable generalization across diverse reasoning tasks [1, 2, 3, 4, 5]. Motivated by this success, the robotics community is currently pursuing foundation models for generalist robot policies capable of flexible and robust decision-making across a wide range of tasks [6, 7, 8], leading to significant industrial investments in large-scale robot learning [9]. However, the pursuit for generalist robot policies remains constrained by the limited availability of high-quality datasets, especially for contact-rich robotic manipulation. Existing datasets [7, 10, 11, 12] are orders of magnitude smaller than those used to train foundation models in other domains, such as Large Language Models (LLMs). The scarcity of diverse, high-fidelity manipulation data limits policy generalization across different embodiments, task contexts, and physical conditions.

To address data scarcity, robot learning researchers often rely on a spectrum of data sources varying in cost, quality, and transferability. The most informative data typically consists of high-quality demonstrations specific to the task, environment, and embodiment [7, 10], but such data is costly and time-consuming to collect, as it requires human teleoperation with specialized hardware. At the opposite end of the spectrum, there is a wealth of lower-quality data in the form of internet videos showing humans and robots performing manipulation tasks [13, 14, 15, 16]. However, the significant embodiment gap and limited action labeling make this data difficult to transfer effectively to robot policies. Simulation data offers a middle ground, providing the potential to generate large, diverse, and high-quality datasets at relatively low cost [17, 18, 19]. In practice, effective policy learning can be achieved by co-training on a mixture of data from different points along this spectrum, reducing data collection costs while improving generalization [20].

A key insight in this work is that human demonstrations and model-based planners complement each other in critical ways for generating high-quality robot data. Human demonstrations, though costly to collect, offer valuable global information for solving complex tasks. However, collecting real-world, contact-rich manipulation data through teleoperation is challenging due to the need for precise multi-contact interactions, which are difficult to achieve in practice due to hardware latency, embodiment mismatches between the human and robot, and the fine-grained control required [21]. In contrast, trajectory optimization has demonstrated success in generating

*Equal Contribution. [1]Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology. [2]Robotics and AI Institute. Correspondence to lujie@mit.edu.

locally-optimal trajectories for contact-rich tasks [22, 23, 24], but often relies on global guidance in the form of good initial guesses.

In this work, we propose a data generation framework that leverages the strengths of both approaches: human demonstrations can provide global guidance, while trajectory optimization can locally refine these demonstrations to ensure dynamic feasibility. Starting with a small number of human demonstrations collected in a virtual reality (VR) environment, our method uses model-based trajectory optimization to generate large datasets of dynamically feasible, contact-rich trajectories in simulation. The demonstrations guide the planner through complex search spaces, while the planner ensures physical consistency and robustness across varying physical parameters and robot embodiments. Our pipeline, visualized in Fig. 1, enables efficient cross-embodiment data transfer, where demonstrations collected with one robot configuration can be adapted to another, and supports domain randomization for improved generalization and robustness. Additionally, it provides the potential to revive and adapt legacy datasets collected with different hardware or configurations, making old datasets valuable for new robot systems.

Our key contributions include:

1) We present an intuitive, embodiment-flexible demonstration interface based on virtual reality and physics simulation, enabling fast data collection for dexterous contact-rich manipulation.
2) We propose a scalable framework that leverages trajectory optimization to transform a small number of human demonstrations into large-scale, physically consistent datasets, enabling generalization across embodiments, initial conditions, and physical parameters.
3) We validate our approach by training policies on the generated dataset for challenging contact-rich manipulation tasks across multiple robot platforms, including bimanual robot arms and a floating base Allegro hand.
4) We achieve high success rates in zero-shot hardware deployment on bimanual iiwa arms, highlighting the utility of augmented datasets in real-world scenarios.

## II. Related Works

In this section, we review the most relevant approaches for generating diverse robot data for contact-rich tasks. We categorize the methods into data collection, data augmentation, model-based planning, demonstration-guided reinforcement learning and cross-embodiment transfer.

### A. Data Collection for Imitation Learning

Behavior Cloning [25], which trains robot policies to mimic expert behavior, has shown impressive empirical results in a wide range of dexterous manipulation tasks [26]. Collecting high-quality robot data has been an essential component of imitation learning (IL). Many such methods rely on human experts teleoperating a robot to accomplish specific tasks. Researchers have adopted interfaces such as 3D spacemouse [26, 27], and puppeteering platforms [28, 29] for end-effector [7] and whole-body control [30, 31].

Virtual and augmented reality (VR/AR) interfaces have recently gained traction as effective alternatives for robot data collection [32, 33], reducing cognitive load, physical strain, and user frustration compared to traditional techniques like kinesthetic teaching or 3D mouse control [34]. These technologies offer a more intuitive data collection paradigm for complex tasks, especially in dexterous manipulation. AR2-D2 [35] enables data collection without a physical robot by projecting a virtual robot into the physical workspace, but lacks real-time feedback necessary for precise control. DART [36] supports data collection entirely in simulation, visualized through a VR headset, but faces challenges bridging the sim-to-real gap for physical robot deployment. ARCap [37] integrates real-time AR feedback, but requires specialized hardware, including an RGBD camera, motion capture gloves, and VR controllers, in addition to the AR headset. ARMADA [38] enables real-world manipulation data collection with bare hands through real-time virtual robot feedback, achieving high success rates when replayed on physical hardware. In contrast to these existing systems, our work focuses on scalable data generation from a small number of human demonstrations by leveraging trajectory optimization, facilitating generalization across different robot embodiments, initial conditions, and physical parameters.

### B. Data Augmentation

Despite many research efforts, collecting large datasets remains time-consuming and costly, requiring a large amount of human effort and resources. To address these challenges, significant effort has been devoted to automating the data generation process through data augmentation techniques. Existing approaches have leveraged state-of-the-art generative models for visual [39, 40, 41] and semantic [42, 43, 44] augmentations. MimicGen [45] and its bimanual extension DexMimicGen [46] automatically synthesize large-scale datasets from a small number of human demonstrations. These works decompose long-horizon tasks into object-centric subtasks and replay transformed demonstrations open loop in simulation. SkillMimicGen [47] extends this paradigm by segmenting tasks into motion and skill components, augmenting local manipulation skills with MimicGen-style replay and using motion planning to connect these skill segments. RoboCasa [48] leverages generative models to create diverse kitchen scenes with abundant 3D assets and utilizes MimicGen for automated trajectory generation. While these approaches have shown success in automating data generation, they primarily rely on kinematic replay of demonstrations, which is often inadequate for contact-rich manipulation tasks. Our work can be viewed as an important extension to MimicGen line of works to support dynamically feasible contact-rich data generation, which requires fine-grained control of the robot and continuous reasoning about making and breaking contacts with the environment.

### C. Trajectory Optimization for Contact-Rich Tasks

Planning and control through contact remains a significant challenge for both learning-based and model-based methods due to the explosion of contact modes and the nonsmooth nature of contact dynamics. To tackle these challenges, researchers have explored various trajectory optimization formulations for multi-contact interactions.

**Contact-Implicit Trajectory Optimization** Existing works based on contact-implicit trajectory optimization (CITO) [23, 22] have sought to formulate the combinatorial problem into a smooth optimization problem by using complementarity constraints. CITO has been applied in various domains, including planar manipulation [49, 50], dynamic pushing [51], and locomotion tasks [52, 53, 54]. Recent efforts have extended CITO for real-time applications as model predictive control (MPC) [55, 56], with successful hardware deployment on quadrupeds using tailored solvers [57, 58]. Aydinoglu et al. [59] parallelize the solution of linear complementarity problems using alternating direction method of multipliers (ADMM) and validate the method on hardware for multi-contact manipulation tasks. A new line of work explores efficient global optimization for contact-rich trajectory optimization [60], but does not yet scale to the tasks we consider here. While CITO shows promising scalability for handling contact modes, it suffers from poor global exploration and relies on good initial guesses [61].

**Sampling-Based Planning** Sampling-based methods have also shown great promise for solving trajectory optimization for contact-rich tasks. Hämäläinen et al. [62] employ sampling-based belief propagation for humanoid balancing, juggling and locomotion. Carius et al. [63] extend the path integral formulation to handle state-input constraints and validate the approach on quadruped stabilization on hardware. More recently, Pezzato et al. [64] applied sampling-based predictive control (SPC) for simpler contact tasks like pushing, while Howell et al. [65] and Li et al. [66] extended SPC to more complex, contact-rich tasks such as in-hand cube reorientation. Pang et al. [67] use smoothed contact dynamics with global sampling to generate contact-rich plans in under a minute, with performance comparable to reinforcement learning. Cheng et al. introduce HiDex [68], a hierarchical planner that combines Monte-Carlo Tree Search with integrated contact projection, achieving rapid planning for dexterous manipulation tasks. Interestingly, directly applying sampling-based planners for contact-rich data generation in behavior cloning can be problematic, as the high entropy of the generated trajectories often degrades downstream policy performance [69, 70].

In this work, we leverage low-entropy human demonstrations to guide the global planning for multi-contact interactions and utilize trajectory optimization to locally refine the trajectories for specific physical parameters and robot embodiments. From a small number of demonstrations, the model-based planner can efficiently generate abundant, high-quality, contact-rich data for training robust robot policies.

### D. Demonstration-Guided Reinforcement Learning

While IL often demands a large number of expert demonstrations to achieve robust and high-performing policies, reinforcement learning (RL) aims to solve tasks autonomously through reward-driven exploration. However, pure RL can suffer from inefficient exploration and the need for extensive reward shaping, especially in complex manipulation tasks [71, 72]. To address these challenges, researchers have explored using demonstrations to guide RL, improving both sample efficiency and exploration quality.

Demonstrations have been integrated into RL pipelines in various ways, including adding them directly to the replay buffer [73, 74], using behavior cloning for policy pretraining [75, 76, 77], and augmenting task rewards with information extracted from demonstrations [78, 79, 80]. Sleiman et al. [81] guide RL with demonstrations generated from a model-based trajectory optimizer for multi-contact loco-manipulation tasks, and validate their method on hardware with a quadrupedal mobile manipulator. While these approaches search over the parameters of a neural network policy and potentially optimize a more global objective, we leverage trajectory optimization as a complementary tool to *locally* refine and expand demonstration trajectories. This enables the efficient generation of contact-rich data while avoiding the computational overhead, approximation errors, and unnecessary exploration associated with RL's high-dimensional search space.

### E. Cross-Embodiment Generalization

Reusing datasets and policies across different embodiments unlocks the potential for large-scale robot learning. One line of work learns latent plans from videos of humans interacting with the environment and transfers this knowledge for robotic manipulation [82, 83]. Another approach involves portable data collection tools, such as hand-held grippers [21, 84], for in-the-wild human demonstrations. While these methods enable policy deployment on multiple robot platforms, they are often constrained to robots with the same end-effector used during data collection, limiting generalization across platforms. On the other hand, to leverage large-scale datasets, recent works pull data from a heterogeneous set of robots ranging from navigation to manipulation, and train a robotic foundation model capable of accomplishing a diverse range of tasks [85, 86]. Our proposed framework enables reusing the same set of easy-to-collect demonstrations for multiple robots, avoiding the need to collect embodiment-specific data for contact-rich tasks.

## III. Data Collection

We present a Virtual Reality (VR)-based data collection pipeline designed for intuitive and efficient collection of human demonstrations across multiple robot embodiments. The pipeline emphasizes simplicity and cross-embodiment generalization while minimizing the reliance on physical robot hardware. While we consider the data collection pipeline to be one of our contributions, we emphasize that the simulation-based large-scale data generation method presented in the

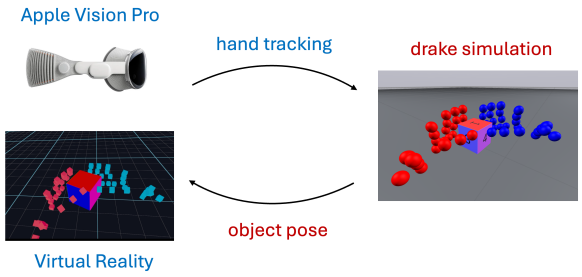


Fig. 2: VR-based human-hand demonstration framework.

next section is independent of this particular data collection approach.

Our data collection pipeline (Fig. 2) is a human-hand demonstration interface in VR. We use an Apple Vision Pro to track the poses of the human demonstrator's hands and stream the poses to the Drake physics simulator [87], which simulates the contact interaction between the object and the hands. The updated object pose is then sent back to Apple Vision Pro for real-time visualization in VR using Vuer [88].

Our demonstration interface is fast and cost-effective. Since the system operates entirely in simulation, it removes the dependency on robot hardware, significantly reducing the cost and complexity of data collection. In practice, it takes approximately 7 minutes to collect 24 long-horizon demos for each considered system. The setup is also intuitive to use, as the human demonstrator does not have to mentally close the embodiment gap between the human body and the specific robot.

We demonstrate our pipeline on two different classes of robot embodiments: a dexterous hand and a bimanual manipulation setup.

**Floating Allegro Hand** For the dexterous hand, we consider a 22-DOF free-floating Allegro hand manipulating a cube on a table as shown in Fig. 3. Since the Allegro hand only has four fingers, we restrict the VR-based demonstrations to using four fingers on the right hand to interact with the object in simulation.

**Bimanual Robot Arms** For the bimanual manipulation setup, we consider two different fixed-base bimanual manipulators: a pair of 7-DOF Kuka LBR iiwa arms, and a pair of Franka Emika Panda arms. Each pair of arms collaboratively manipulates a big box (Fig. 3). During the VR demonstrations, the human demonstrator uses both index fingers to manipulate a small cube in VR and constrains their wrist movement to mimic the fixed base. During kinematic motion retargeting (detailed in Sec. IV-A), the small cube and fingers are scaled to match the size of the larger box and the robot manipulators.

This design facilitates two forms of cross-embodiment generalization. First, it utilizes easy-to-collect human finger demonstrations to guide planning for harder and higher-dimensional tasks, such as the dual-arm manipulators. Second, it supports the reuse of the same set of demonstrations across multiple robot platforms, as both the iiwa and Panda arms can leverage the same data to accomplish the manipulation task, eliminating the need for embodiment-specific demonstrations.

## IV. Automated Data Generation

In this section, we present our method for automatically generating large quantities of physically feasible trajectories for contact-rich manipulation tasks across a range of objects, initial conditions, and embodiments from only a handful of demonstrations. The presented method also offers the potential to adapt legacy datasets collected using outdated configurations to new robot settings, reducing the cost of collecting large amounts of data on the new robot setups from scratch.

Our method starts out by retargeting kinematic motions from the original embodiment-flexible human demonstrations collected in VR to the specific robot embodiment in simulation, producing kinematically feasible trajectories. These trajectories are then refined and augmented through the use of local trajectory optimization to obtain dynamically feasible trajectories for a range of physical parameters. The following subsections provide a detailed breakdown of each step in the pipeline.

### A. Kinematic Motion Retargeting

Given a sequence of demonstrations $x^{\text{demo}}_{0:T}$ with horizon $T$, we aim to find the robot configurations $q^{\text{retarget}}_{0:T}$ that match the positioning of the demonstrator while avoiding penetration and obeying joint limits. At each time step, we solve the following nonconvex program:

$$q_t^{\text{retarget}\star} = \arg\min_{q_t^{\text{retarget}}} \quad \sum_{i=0}^{N} w_i \|\psi_i(q_t^{\text{retarget}}) - \tilde{\psi}_i(x_t^{\text{demo}})\|^2 \quad \text{(1a)}$$

$$\text{s.t.} \quad \phi_j(q_t^{\text{retarget}}) \geq 0, \ \forall j \quad \text{(1b)}$$

$$q_{\min} \leq q_t^{\text{retarget}} \leq q_{\max}, \quad \text{(1c)}$$

where $w_i > 0$ are weight parameters, and $\psi_i$ and $\tilde{\psi}_i$ represent the $i$-th mappings from the robot configuration and demonstrator state to corresponding points on the embodiments. The corresponding points of interest for each robot/demonstrator pair are manually defined. For example, on the bimanual robot arm system, $\psi_0$ is the forward kinematics from the robot joint angles to the left robot arm's end effector position, while $\tilde{\psi}_0$ is a map from the hand pose to the fingertip of the left index finger. We find the resulting plans generated by trajectory optimization relatively robust to the correspondence and weight parameter selection. $\phi_j$ denotes the signed distance function between the $j$-th collision pair and (1b) enforces non-penetration constraints. $q_{\min}$ and $q_{\max}$ are the lower and upper bounds on the joint angles. Notice that $q^{\text{retarget}}$ and $x^{\text{demo}}$ can have different dimensions as long as both $\psi_i$ and $\tilde{\psi}_i$ map them to vectors in the same space (e.g., Apple Vision Pro captures 5 landmarks on the index finger while each robot arm has 7 DOF in the bimanual robot arm system). We solve (1) using a Sequential Quadratic Programming (SQP)-style algorithm: during each iteration, the nonpenetration constraint (1b) is linearized and the matching objective (1a) is quadratically approximated around the solution to the previous iteration. We warmstart the solution of the nonlinear program at time $t$

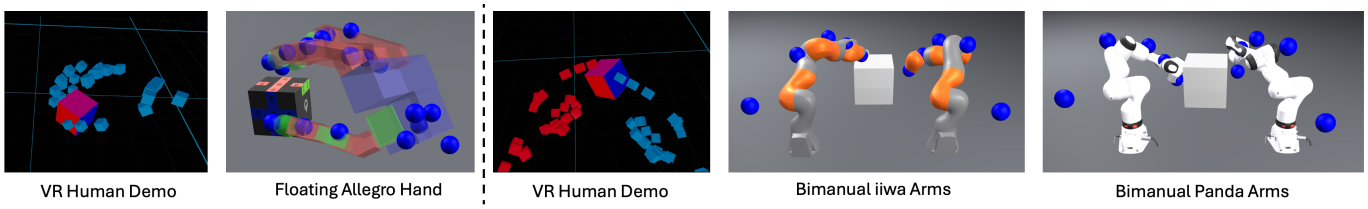


Fig. 3: Human hand demo in VR and kinematic retargeting for different embodiments. The blue spheres illustrate the demo hand landmarks scaled to the specific system.

with the optimal solution from the previous timestep $q_{t-1}^{\text{retarget}\star}$ to encourage faster convergence and temporal consistency.

### B. Demonstration-Guided Trajectory Optimization

The kinematically consistent robot trajectories $q_{0:T}^{\text{retarget}\star}$ are generally not dynamically feasible due to the embodiment gap and differences in physical parameters. However, they can provide good guidance on generating dynamically feasible trajectories with complex multi-contact interactions. In particular, human demonstrations provide global information about when and where to make contact with the object, which model-based planning can then locally refine. We define the retargeted system state $x_t^{\text{retarget}}$ to incorporate both the object state $x_t^{\text{object}}$, which is a subset of $x_t^{\text{demo}}$, and the robot state as a function of $q_t^{\text{retarget}\star}$. The trajectory $x_{0:T}^{\text{retarget}}$ is then locally refined by solving the following nonconvex optimization program:

$$
\begin{aligned}
x_t^\star, u_t^\star = \arg\min_{x_t, u_t} \quad & \|x_T - x_T^{\text{retarget}}\|_{Q_T}^2 + \\
& \sum_{t=0}^{T-1} (\|x_t - x_t^{\text{retarget}}\|_{Q_t}^2 + \|u_t\|_{R_t}^2) && \text{(2a)} \\
\text{s.t.} \quad & x_{t+1} = f(x_t, u_t) && \text{(2b)} \\
& \phi_j(x_t) \geq 0, \ \forall j && \text{(2c)} \\
& x_{\min} \leq x_t \leq x_{\max} && \text{(2d)} \\
& u_{\min} \leq u_t \leq u_{\max}. && \text{(2e)}
\end{aligned}
$$

Here, $f$ is obtained by time-stepping the dynamics engine, $x_{\min}/x_{\max}$ ($u_{\min}/u_{\max}$) are the lower and upper bounds on the state (input), $Q_t, R_t$ are the cost matrices for the state and input, respectively, and $Q_T$ is the cost matrix for the terminal state. To encourage precise tracking of the object trajectory, we assign higher weights to the entries of $Q_t$ which correspond to $x_t^{\text{object}}$. The detailed parameters can be found in Appendix IX-A.

In general, model-based planners can struggle to discover high-quality long-horizon contact-rich trajectories without demonstrations. CITO requires good initial guesses and can easily get stuck in local optima without making progress. Human demonstrations offer valuable global guidance that helps overcome these challenges, and $x_{0:T}^{\text{retarget}}$ can naturally serve as the initial guess to CITO-based methods where local adjustments are made to obey dynamical constraints (2b).

Thanks to access to the system dynamics $f$ in simulation, we can locally perturb the physical parameters as well as robot and object states around a nominal demonstration. From the single demonstration, we can solve (2) for a distribution of tasks with different dynamics $f(x_t, u_t, \theta_t)$, where $\theta_t \sim \rho$ represents all the perturbations. We assume the kinematically retargeted trajectory $x_{0:T}^{\text{retarget}}$ still provides good guidance on achieving the task in the vicinity of the nominal demonstration. This way, a large number of physically consistent trajectories with various physical properties and initial conditions can be generated from a single human demonstration. We outline our data generation pipeline in Algorithm 1.

**Algorithm 1: Automated Data Generation**

**Input:** Probability distribution $\rho$, augmentation number $N$, demo trajectory $x_{0:T}^{\text{demo}}$;
**Output:** $N$ dynamically consistent trajectories on target embodiments $\{(x_{0:T}^\star, u_{0:T-1}^\star)\}$;
$q_{0:T}^{\text{retarget}\star} \leftarrow$ Solve (1) for $x_{0:T}^{\text{demo}}$;
**for** $n = 1, \ldots, N$ **do**
  Sample $\theta_{0:T} \sim \rho$;
  $(x_{0:T}^\star, u_{0:T-1}^\star) \leftarrow$ Solve (2) with $x_{0:T}^{\text{retarget}}, \theta_{0:T}$ and $x_{t+1} = f(x_t, u_t, \theta_t)$;

## V. Trajectory Optimization Experiments

While kinematic retargeting of demonstrations might suffice to generate data for simpler manipulation tasks such as pick and place, it often falls short for the more challenging contact-rich tasks requiring frequent contact mode switches and fine-grained actions. In this section, we demonstrate that trajectory optimization is crucial for generating diverse, dynamically feasible contact-rich trajectories on three high-dimensional dexterous manipulation systems: a floating Allegro hand, bimanual iiwa arms, and bimanual Panda arms.

Our data generation framework is agnostic to the choice of the trajectory optimizer. We implement the cross-entropy

| Parameter | Floating Allegro Hand | Bimanual Robot Arms |
|---|---|---|
| Init. obj. trans. pert. (cm) | [±1.5, ±1.5, 0] | [±5, ±5, 0] |
| Init. obj. rot. pert. (rad) | [0, 0, ±0.3] | [0, 0, ±0.3] |
| Object side length (cm) | [5.8, 6.2] | [28, 32] |
| Object mass (kg) | [0.1, 0.3] | [0.25, 0.75] |
| Friction coefficients | [0.7, 1.3] | [0.2, 0.4] |
| Task horizon (s) | 25 | 50 / 260 (Panda / iiwa) |

TABLE I: Ranges of different physical parameters $\theta$. The initial object pose is only perturbed in yaw, x, and y to ensure the object sits stably on the table.

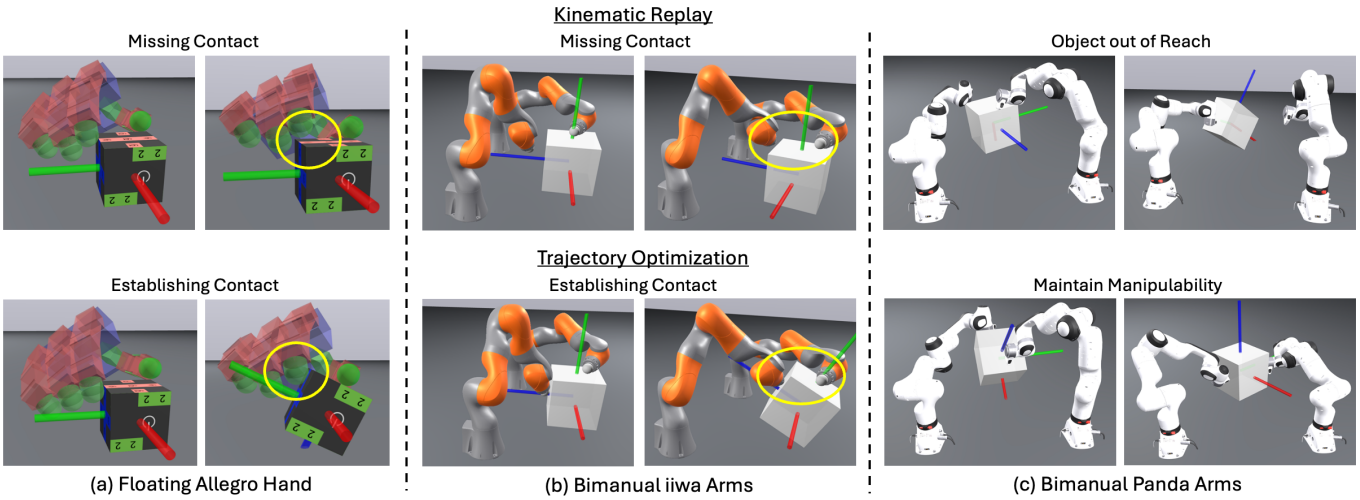


Fig. 4: **Trajectory optimization is crucial for generating dynamically feasible trajectories**. (Top) Before trajectory optimization, the kinematically retargeted demos easily lose contact and drive the object out of reach with different physical parameters or slight deviations in object states. (Bottom) Trajectory optimization encourages robots to establish contact with and maintain good manipulability of the object. The tricolor axis indicates the object orientation.

| Perturbation | Allegro Hand | iiwa Arms | Panda Arms |
|---|---|---|---|
| Original demo | 4 / 24 | 5 / 24 | 6 / 24 |
| Object size | 2 / 24 | 1 / 24 | 4 / 24 |
| Initial object translation | 1 / 24 | 3 / 24 | 2 / 24 |
| Initial object orientation | 2 / 24 | 3 / 24 | 3 / 24 |
| Trajectory optimization | 2164 / 3000 | 2252 / 3000 | 2462 / 3000 |

TABLE II: Success rates of replaying kinematically retargeted trajectories of the 24 original human demos, and trajectory optimization under random perturbations in physical parameters and object initial conditions.

method (CEM) [89] to solve (2) over a distribution of physical parameters and initial conditions, as specified in Table I.

**Task** Manipulating the object to a target pose on the table (Fig. 6). The object is initially placed randomly on the table with an arbitrary face upward. Task success is defined as the object reaching within 3 cm and 0.2 rad of the target pose for the Allegro hand, and within 10 cm and 0.2 rad for the bimanual robot arms. This task requires long-horizon reasoning of complex multi-contact interactions between the robot and the object. The necessary frequent contact mode switches and high-dimensional action space pose great challenges for traditional model-based planners, while the precise contact interactions require fine-grained control actions.

**Dynamic Feasibility** While kinematic motion retargeting can generate visually plausible robot and object trajectories, these trajectories often lack dynamical consistency due to the differences in physical parameters and embodiment between the human demonstrator and the target robot. To illustrate this, we replay the kinematically retargeted trajectories of the original 24 human demos and record the success rates for each system in Table II. Furthermore, we randomly sample object sizes and perturbations of initial object poses according to Table I and roll out the nominal kinematically retargeted trajectories. Some trajectories still succeed under certain perturbations thanks to caging grasps or other strategies that encourage robustness during the human demonstration. For all the systems, the successful rollouts are relatively short, manipulating the object to the goal pose within only 1 or 2 rotations.

The low success rate of purely kinematically retargeted trajectories highlights the importance of trajectory optimization for locally refining the demos for the particular embodiments and physical parameters. Before trajectory optimization, the floating Allegro hand lightly touches the cube and easily loses contact when rotating it clockwise (demonstrated in Fig. 4a). After trajectory optimization, the hand increases the contact area, establishing a stable grip for rotation. In Fig. 4b, similar behavior that encourages contact can be observed for the bimanual iiwa arms: the demo trajectory tries to rotate the box clockwise only using a single arm, while trajectory optimization encourages the other arm to help hold the box and reorient the box more stably. These refinements that encourage contact are particularly helpful when the object is heavier or smaller, or when the friction coefficients are lower than expected. In addition, replaying the kinematically retargeted trajectory often fails when the object pose deviates slightly from the demonstration, driving the object out of reach (visualized in Fig. 4c). In contrast, trajectory optimization accounts for the system's true dynamics and can adjust the robot's actions accordingly. The success rates of trajectory optimization under random perturbations in physical parameters and object initial conditions for each system are recorded in Table II.

**Cross-Embodiment Generalization** We demonstrate that a single set of human demonstrations can be effectively repurposed to generate dynamically consistent, contact-rich trajectories across different robotic embodiments with varying task horizons. Specifically, human demonstrations involving two index fingers manipulating a small cube are retargeted

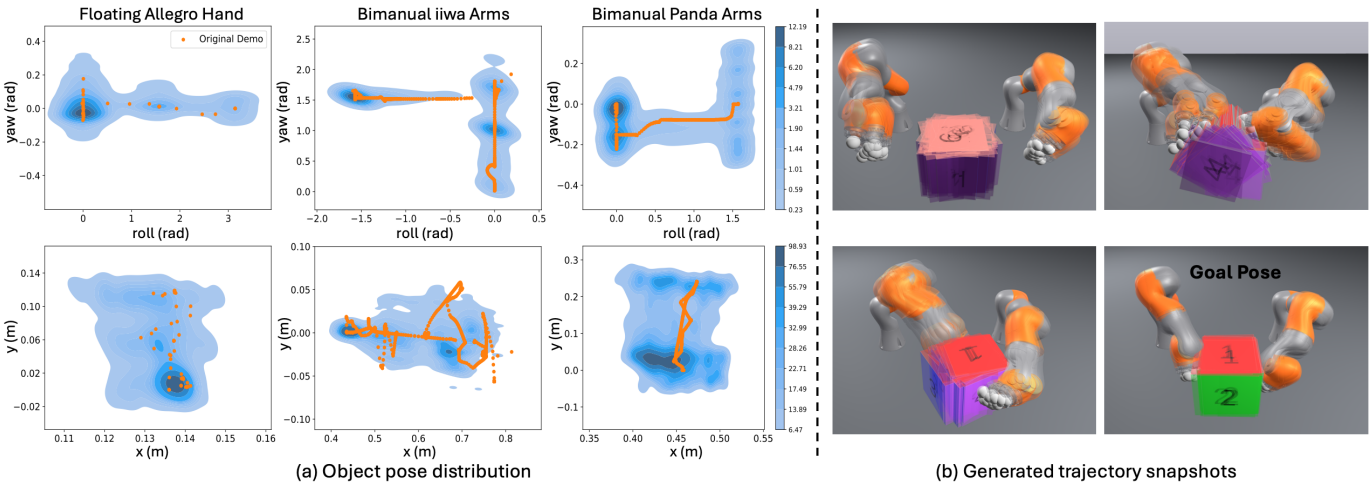


Fig. 5: **Distribution and snapshots of trajectories generated from a single demonstration.** (a) The original demonstration (orange) is locally perturbed and augmented to about 100 dynamically feasible contact-rich trajectories (blue) for each system. The density map represents the object pose distribution of the generated trajectories in the specific 2-dimensional slices. (b) Snapshots of 30 dynamically feasible trajectories under random physical parameters and object initial poses for bimanual iiwa arms are visualized.

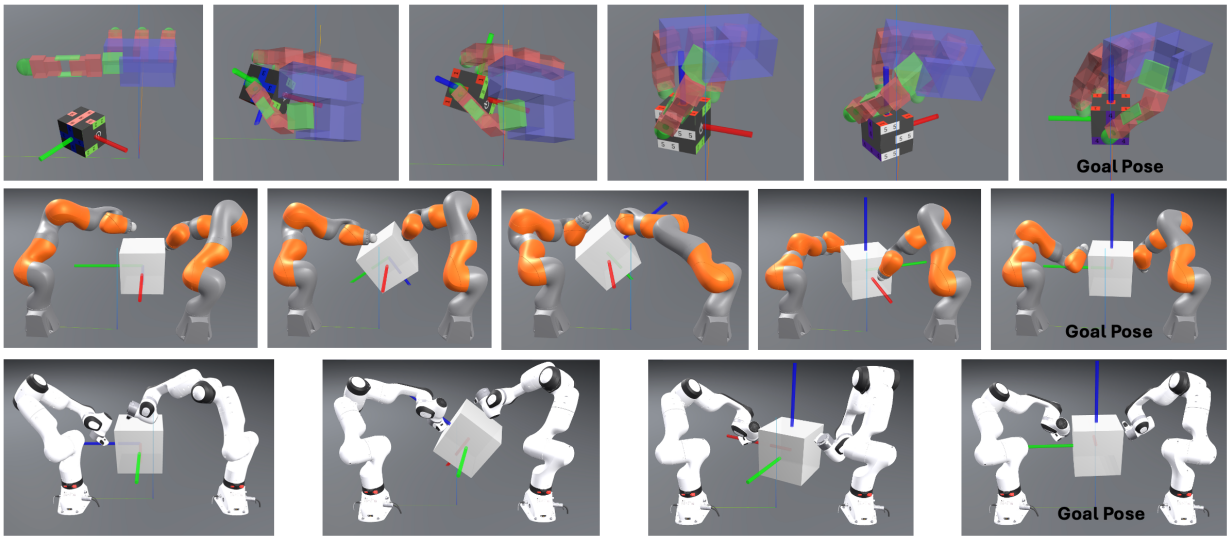


Fig. 6: **Policy rollouts for different embodiments.** The object manipulation task requires the robots to frequently make and break contact with the object. It also requires precise control of the robot since small deviations in positions can result in missing contact interactions and lead to task failure.

to fixed-base bimanual Kuka LBR iiwa and Franka Emika Panda arms manipulating a larger box (visualized in Fig. 3). This approach addresses key challenges in data collection for contact-rich tasks: directly teleoperating two real robot arms to flip a large box would be both physically demanding and cost-prohibitive due to hardware latency, limited feedback, and the embodiment gap–differences in kinematic structure, degrees of freedom, and workspace between human and robotic arms. In contrast, performing the same task on a smaller scale using human fingers is more intuitive, reduces physical effort, and enables faster, more consistent demonstration collection.

The iiwa and Panda arms differ in contact geometry, velocity limits, and joint constraints, all of which are explicitly modeled within the trajectory optimization framework described in (2). For safe hardware deployment, we enforce conservative velocity limits on the iiwa arms, while only applying soft velocity regularization on the Panda arms in simulation to allow for more aggressive motions.

**Data Diversity** Trajectory optimization efficiently augments a single demonstration to a wide distribution of trajectories with locally perturbed physical parameters and initial conditions as visualized in Fig. 5. The diverse states in the generated dataset cover a larger training distribution and encourage smoother learned policies, as will be discussed in the next section.

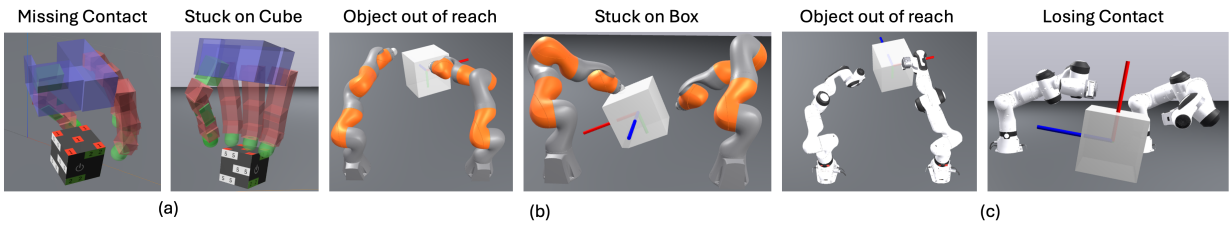


Fig. 7: **Failure cases of baselines.** (a) The baseline policy trained on the original 24 demonstrations for the floating Allegro hand frequently misses contact or gets stuck on the cube. (b-c) The baseline policies for the bimanual robot arms often exhibit jittery motion, resulting in loss of contact, the box being kicked out of reach, or the robot arms running into and getting stuck on the box surface.

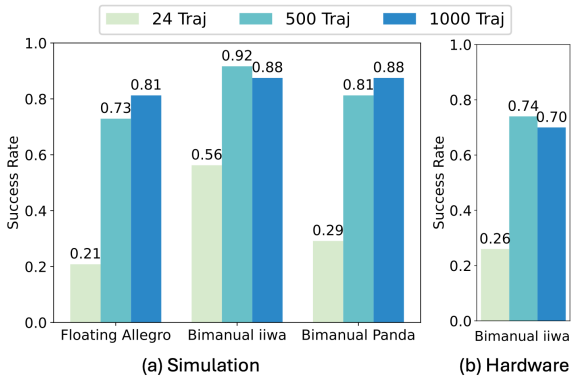


Fig. 8: Success rates of policy evaluation in simulation and hardware.

## VI. Behavior Cloning Experiments

We illustrate our framework's capability to efficiently produce diverse, high-quality contact-rich datasets for training behavior cloning policies across multiple robotic platforms, including the floating Allegro hand and the bimanual Panda arms in simulation as well as bimanual iiwa arms on hardware. We show that policies trained on the generated data generalize to a wide distribution of physical parameters and initial conditions, and are much more robust and performant than the ones trained only on the original demonstrations.

### A. Policy Evaluation in Simulation

From only 24 human demonstrations, our data generation pipeline can efficiently generate thousands of dynamically feasible contact-rich trajectories using trajectory optimization. We train state-based diffusion policies [26] on the 24 original demo trajectories, as well as 500 and 1000 generated trajectories. While our method is compatible with any Behavior Cloning algorithm, we adopt diffusion policies due to its recent success in contact-rich tasks [21, 70, 90]. Fig. 6 visualizes the policy rollouts. We evaluate the performance by conducting 48 policy rollouts for each embodiment in simulation and record the success rates in Fig. 8. The success criteria are the same as specified in the trajectory optimization experiments.

*1) Floating Allegro Hand:* While the human demonstrator completes the task in approximately 5 seconds on average in the virtual reality environment, the demonstration trajectories are temporally scaled by a factor of 2.5 to ensure smoother, dynamically feasible motions on the floating Allegro hand, which is subject to velocity limits. We define the task horizon as 25 seconds to allow the policy sufficient time to recover from missed contacts and other errors during the execution. The task complexity arises from the 22-dimensional action space of the Allegro hand and the long-horizon nature of the task, which requires a sequence of coordinated rolling, pitching, and yawing actions to reorient the cube to an upright position. These factors together present significant challenges for traditional model-based planners without guidance.

The baseline behavior cloning policy trained on the original set of 24 demonstrations achieves a success rate of $10/48 = 21\%$ and exhibits significant jittery behavior when encountering out-of-distribution states. The workspace, characterized by diverse object orientations and translations, is sufficiently large that minor deviations during policy rollouts often drive the trajectory out of the demonstrated distribution. Common failure modes include the Allegro hand repeatedly missing contact with the cube or becoming stuck on its surface while attempting reorientation (visualized in Fig. 7a), which often result in the object being trapped in intermediate orientations. In contrast, policies trained on the expanded dataset generated by our pipeline demonstrate a higher likelihood of re-establishing contact with the object after initial misses, resulting in significantly improved success rates up to $39/48 = 81\%$.

*2) Bimanual Robot Arms:* The baseline policy trained on the original set of 24 human demonstrations achieves a success rate of $27/48 = 56\%$ on the bimanual iiwa system. We hypothesize that the restrictive velocity limits encourage more quasi-static behavior, leading to longer trajectories with a higher density of state-action pairs in the training data. In contrast, the baseline policy yields a success rate of $14/48 = 29\%$ on the bimanual Panda system, likely due to the more dynamic nature of the learned behavior under its looser velocity constraints. Both baseline policies exhibit remarkably jittery motion, frequently kicking the box out of reach, losing contact, or running into and getting stuck on the box surface during reorientation (visualized in Fig. 7b and c). Policies trained on the augmented dataset, however, generate significantly smoother trajectories and are capable of re-establishing contact with the object after initial misses, resulting in as high as $44/48 = 92\%$ success rates for bimanual iiwa arms and $42/48 = 87.5\%$ for bimanual Panda arms. Additionally, the learned policies capture multimodal behaviors observed in the original human demonstrations, such as rotating the box either clockwise or counterclockwise for similar object poses.

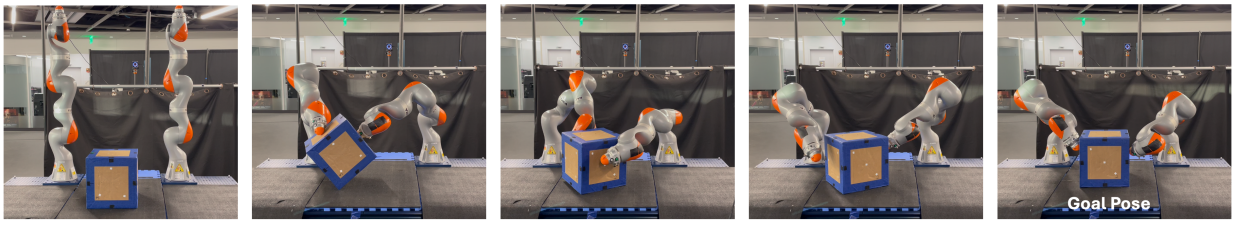


Fig. 9: **Policy rollouts on hardware.** The fixed-base bimanual iiwa arms perform a sequence of coordinated rolling, pitching, and yawing actions to reorient the box to the goal pose.

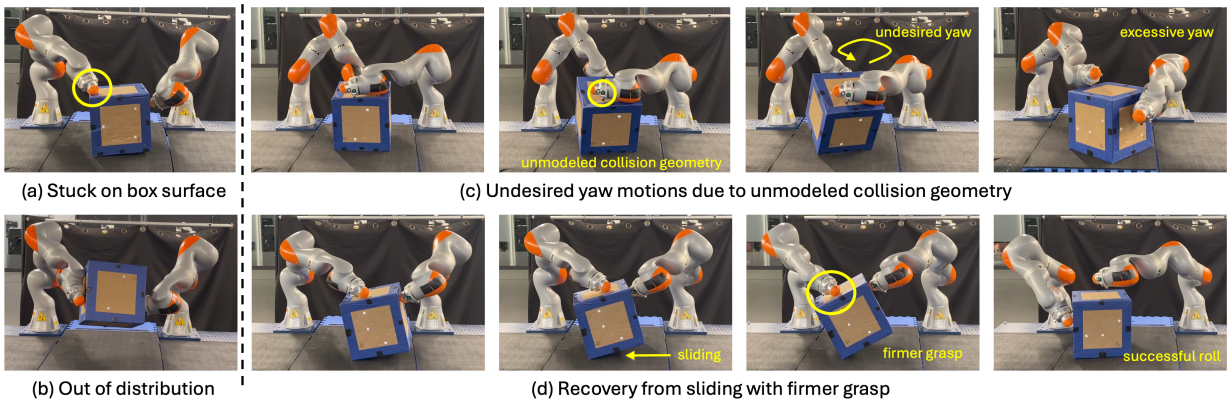


Fig. 10: **Policy failure and recovery on hardware.** The baseline policy frequently (a) gets stuck on the box surface when small deviations from the demonstration trajectories occur, and (b) struggles to recover from out-of-distribution states, where the object is never intentionally lifted for accomplishing the task in the generated dataset. Policies trained on augmented datasets (c) sometimes fail due to unmodeled collision geometry, but (d) can recover from undesired sliding by employing firmer grasps found by trajectory optimization.

### *B. Policy Evaluation on Hardware*

We zero-shot deploy the trained policies on hardware for bimanual iiwa arms to flip a 30 cm cubic box on a table (Fig. 9). An OptiTrack motion capture system is employed to estimate the object pose. The baseline behavior cloning policy only achieves $6/23 = 26\%$ success rate, with most successful rollouts being relatively short-horizon, involving only 1 or 2 rotations. Common failure modes of the baseline policy include: 1) deviation from the demonstration trajectory, causing the arms to collide with the box surface (Fig. 10a), and 2) significant box sliding during rolling, resulting in the policy encountering out-of-distribution states and failing to recover (Fig. 10b). In contrast, as shown in Fig. 8b, the policy trained on 500 generated trajectories achieves $17/23 = 74\%$ success rate, while the policy trained on 1000 generated trajectories achieves $16/23 = 70\%$ success rate. Despite occasional box sliding during rolling, these policies demonstrate an improved ability to stabilize the box by using one arm to hold the opposite side more firmly to prevent further sliding (Fig 10d). However, as visualized in Fig 10c, both policies trained on the augmented datasets exhibit failure modes originating from unmodeled collision geometries on iiwa arms, which lead to significant undesired yaw motions of the box during pitch actions.

## VII. Limitations and Future Work

While our method efficiently generates abundant contact-rich trajectories, several limitations remain. First, although our human-hand demonstration framework is fast and intuitive, it may not fully exploit the kinematic capabilities of the target robot, such as continuous joint rotation or specialized dexterous maneuvers. Future work could explore the application of our automated data generation framework to embodiment-aware legacy datasets, better capturing the unique motion capabilities of different robotic systems.

Second, although our method demonstrates strong performance in the vicinity of the demonstration due to trajectory optimization, the learned policies struggle to recover from states far outside the demonstrated regions, such as those resulting from catastrophic failure. Future work could explore more advanced planning techniques to iteratively improve the learned policies' robustness in unvisited regions of the state space.

Third, we have demonstrated the effectiveness of our pipeline primarily for training robust state-based policies. Extending the framework to train visuomotor policies by incorporating high-quality synthetic rendering from simulation could further improve policy transferability to real-world scenarios.

## VIII. Conclusion

In this work, we present a novel, cost-effective pipeline that combines physics-based simulations, human demonstrations, and model-based planning to address data scarcity in contact-rich robotic manipulation tasks. A key insight of our approach is that human demonstrations—even when collected on a different morphology—offer valuable global task information

that model-based planners often struggle to discover independently due to the high-dimensional search space and complex contact dynamics. By leveraging these demonstrations as a global prior, our method refines and augments them through kinematic retargeting and trajectory optimization, resulting in large datasets of dynamically feasible trajectories across a range of physical parameters, initial conditions, and embodiments. Our framework significantly reduces the reliance on costly, hardware-specific data collection while offering the potential to reuse legacy datasets collected with outdated hardware or configurations. We demonstrate its effectiveness across multiple robotic systems in simulation and successfully zero-shot deploy policies trained on the augmented dataset to a bimanual iiwa hardware setup.

## REFERENCES


[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[2] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[3] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark, *et al.*, "Training compute-optimal large language models," *arXiv preprint arXiv:2203.15556*, 2022.

[4] R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen, *et al.*, "Palm 2 technical report," *arXiv preprint arXiv:2305.10403*, 2023.

[5] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[6] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.

[7] A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models," *arXiv preprint arXiv:2310.08864*, 2023.

[8] O. M. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, T. Kreiman, C. Xu, *et al.*, "Octo: An open-source generalist robot policy," *arXiv preprint arXiv:2405.12213*, 2024.

[9] Reuters, "Robotics startup figure raises $67.5 million from microsoft, nvidia, and other big tech firms," *Reuters*, 2024, accessed: 2024-12-08.

[10] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, *et al.*, "Droid: A large-scale in-the-wild robot manipulation dataset," *arXiv preprint arXiv:2403.12945*, 2024.

[11] H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He, V. Myers, M. J. Kim, M. Du, *et al.*, "Bridgedata v2: A dataset for robot learning at scale," in *Conference on Robot Learning*. PMLR, 2023, pp. 1723–1736.

[12] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and C. Finn, "Robonet: Large-scale multi-robot learning," *arXiv preprint arXiv:1910.11215*, 2019.

[13] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, *et al.*, "Scaling egocentric vision: The epic-kitchens dataset," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 720–736.

[14] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real-world robot learning with masked visual pre-training," in *Conference on Robot Learning*. PMLR, 2023, pp. 416–426.

[15] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.

[16] S. Karamcheti, S. Nair, A. S. Chen, T. Kollar, C. Finn, D. Sadigh, and P. Liang, "Language-driven representation learning for robotics," *arXiv preprint arXiv:2302.12766*, 2023.

[17] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, *et al.*, "Sapien: A simulated part-based interactive environment," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 097–11 107.

[18] G. Brockman, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[19] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, "Rlbench: The robot learning benchmark & learning environment," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3019–3026, 2020.

[20] L. Wang, X. Chen, J. Zhao, and K. He, "Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers," *arXiv preprint arXiv:2409.20537*, 2024.

[21] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots," *arXiv preprint arXiv:2402.10329*, 2024.

[22] I. Mordatch, E. Todorov, and Z. Popović, "Discovery of complex behaviors through contact-invariant optimization," *ACM Transactions on Graphics (ToG)*, vol. 31, no. 4, pp. 1–8, 2012.

[23] M. Posa, C. Cantu, and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *The International Journal of Robotics Research*, vol. 33,

no. 1, pp. 69–81, 2014.

[24] T. A. Howell, K. Tracy, S. Le Cleac'h, and Z. Manchester, "Calipso: A differentiable solver for trajectory optimization with conic and complementarity constraints," in *The International Symposium of Robotics Research*. Springer, 2022, pp. 504–521.

[25] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," *Advances in neural information processing systems*, vol. 1, 1988.

[26] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, p. 02783649241273668, 2023.

[27] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu, "Viola: Imitation learning for vision-based manipulation with object proposal priors," in *Conference on Robot Learning*. PMLR, 2023, pp. 1199–1210.

[28] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," *arXiv preprint arXiv:2401.02117*, 2024.

[29] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, K. Ghasemipour, C. Finn, and A. Wahid, "Aloha unleashed: A simple recipe for robot dexterity," *arXiv preprint arXiv:2410.13126*, 2024.

[30] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn, "Humanplus: Humanoid shadowing and imitation from humans," *arXiv preprint arXiv:2406.10454*, 2024.

[31] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. Kitani, C. Liu, and G. Shi, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," *arXiv preprint arXiv:2406.08858*, 2024.

[32] F. Ebert, Y. Yang, K. Schmeckpeper, B. Bucher, G. Georgakis, K. Daniilidis, C. Finn, and S. Levine, "Bridge data: Boosting generalization of robotic skills with cross-domain datasets," *arXiv preprint arXiv:2109.13396*, 2021.

[33] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 5628–5635.

[34] A. Smith and M. Kennedy III, "An augmented reality interface for teleoperating robot manipulators: Reducing demonstrator task load through digital twin control," *arXiv preprint arXiv:2409.18394*, 2024.

[35] J. Duan, Y. R. Wang, M. Shridhar, D. Fox, and R. Krishna, "Ar2-d2: Training a robot without a robot," *arXiv preprint arXiv:2306.13818*, 2023.

[36] Y. Park, J. S. Bhatia, L. Ankile, and P. Agrawal, "Dexhub and dart: Towards internet scale robot data collection," *arXiv preprint arXiv:2411.02214*, 2024.

[37] S. Chen, C. Wang, K. Nguyen, L. Fei-Fei, and C. K. Liu, "Arcap: Collecting high-quality human demonstrations for robot learning with augmented reality feedback," *arXiv preprint arXiv:2410.08464*, 2024.

[38] N. Nechyporenko, R. Hoque, C. Webb, M. Sivapurapu, and J. Zhang, "Armada: Augmented reality for robot manipulation and robot-free data acquisition," *arXiv preprint arXiv:2412.10631*, 2024.

[39] X. Zhang, M. Chang, P. Kumar, and S. Gupta, "Diffusion meets dagger: Supercharging eye-in-hand imitation learning," *arXiv preprint arXiv:2402.17768*, 2024.

[40] S. Tian, B. Wulfe, K. Sargent, K. Liu, S. Zakharov, V. Guizilini, and J. Wu, "View-invariant policy learning via zero-shot novel view synthesis," *arXiv preprint arXiv:2409.03685*, 2024.

[41] L. Y. Chen, C. Xu, K. Dharmarajan, Z. Irshad, R. Cheng, K. Keutzer, M. Tomizuka, Q. Vuong, and K. Goldberg, "Rovi-aug: Robot and viewpoint augmentation for cross-embodiment robot learning," in *Conference on Robot Learning (CoRL)*, 2024.

[42] Z. Mandi, H. Bharadhwaj, V. Moens, S. Song, A. Rajeswaran, and V. Kumar, "Cacti: A framework for scalable multi-task multi-scene visual imitation learning," *arXiv preprint arXiv:2212.05711*, 2022.

[43] Z. Chen, S. Kiami, A. Gupta, and V. Kumar, "Genaug: Retargeting behaviors to unseen situations via generative augmentation," *arXiv preprint arXiv:2302.06671*, 2023.

[44] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, J. Peralta, B. Ichter, *et al.*, "Scaling robot learning with semantically imagined experience," *arXiv preprint arXiv:2302.11550*, 2023.

[45] A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, Y. Zhu, and D. Fox, "Mimicgen: A data generation system for scalable robot learning using human demonstrations," *arXiv preprint arXiv:2310.17596*, 2023.

[46] Z. Jiang, Y. Xie, K. Lin, Z. Xu, W. Wan, A. Mandlekar, L. Fan, and Y. Zhu, "Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning," *arXiv preprint arXiv:2410.24185*, 2024.

[47] C. Garrett, A. Mandlekar, B. Wen, and D. Fox, "Skillmimicgen: Automated demonstration generation for efficient skill learning and deployment," in *8th Annual Conference on Robot Learning*, 2024.

[48] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu, "Robocasa: Large-scale simulation of everyday tasks for generalist robots," *arXiv preprint arXiv:2406.02523*, 2024.

[49] A. Ö. Önol, P. Long, and T. Padır, "Contact-implicit trajectory optimization based on a variable smooth contact model and successive convexification," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 2447–2453.

[50] J. Moura, T. Stouraitis, and S. Vijayakumar, "Non-prehensile planar manipulation via trajectory optimization with complementarity constraints," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 970–976.

[51] J.-P. Sleiman, J. Carius, R. Grandia, M. Wermelinger, and M. Hutter, "Contact-implicit trajectory optimization

for dynamic object manipulation," in *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2019, pp. 6814–6821.

[52] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 4906–4913.

[53] M. Neunert, F. Farshidian, and J. Buchli, "Efficient whole-body trajectory optimization using contact constraint relaxation," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 43–48.

[54] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, "Gait and trajectory optimization for legged systems through phase-based end-effector parameterization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1560–1567, 2018.

[55] M. Wang, A. Ö. Önol, P. Long, and T. Padır, "Contact-implicit planning and control for non-prehensile manipulation using state-triggered constraints," in *The International Symposium of Robotics Research*. Springer, 2022, pp. 189–204.

[56] V. Kurtz, A. Castro, A. Ö. Önol, and H. Lin, "Inverse dynamics trajectory optimization for contact-implicit model predictive control," *arXiv preprint arXiv:2309.01813*, 2023.

[57] M. Neunert, M. Stäuble, M. Giftthaler, C. D. Bellicoso, J. Carius, C. Gehring, M. Hutter, and J. Buchli, "Whole-body nonlinear model predictive control through contacts for quadrupeds," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1458–1465, 2018.

[58] S. Le Cleac'h, T. A. Howell, S. Yang, C.-Y. Lee, J. Zhang, A. Bishop, M. Schwager, and Z. Manchester, "Fast contact-implicit model predictive control," *IEEE Transactions on Robotics*, 2024.

[59] A. Aydinoglu, A. Wei, W.-C. Huang, and M. Posa, "Consensus complementarity control for multi-contact mpc," *IEEE Transactions on Robotics*, 2024.

[60] B. P. Graesdal, S. Y. C. Chia, T. Marcucci, S. Morozov, A. Amice, P. A. Parrilo, and R. Tedrake, "Towards tight convex relaxations for contact-rich manipulation," *arXiv preprint arXiv:2402.10312*, 2024.

[61] H. T. Suh, T. Pang, T. Zhao, and R. Tedrake, "Dexterous contact-rich manipulation via the contact trust region," 2025.

[62] P. Hämäläinen, J. Rajamäki, and C. K. Liu, "Online control of simulated humanoids using particle belief propagation," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, pp. 1–13, 2015.

[63] J. Carius, R. Ranftl, F. Farshidian, and M. Hutter, "Constrained stochastic optimal control with learned importance sampling: A path integral approach," *The International Journal of Robotics Research*, vol. 41, no. 2, pp. 189–209, 2022.

[64] C. Pezzato, C. Salmi, M. Spahn, E. Trevisan, J. Alonso-Mora, and C. H. Corbato, "Sampling-based model predictive control leveraging parallelizable physics simulations," *arXiv preprint arXiv:2307.09105*, 2023.

[65] T. Howell, N. Gileadi, S. Tunyasuvunakool, K. Zakka, T. Erez, and Y. Tassa, "Predictive sampling: Real-time behaviour synthesis with mujoco," *arXiv preprint arXiv:2212.00541*, 2022.

[66] A. H. Li, P. Culbertson, V. Kurtz, and A. D. Ames, "Drop: Dexterous reorientation via online planning," *arXiv preprint arXiv:2409.14562*, 2024.

[67] T. Pang, H. T. Suh, L. Yang, and R. Tedrake, "Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models," *IEEE Transactions on robotics*, 2023.

[68] X. Cheng, S. Patil, Z. Temel, O. Kroemer, and M. T. Mason, "Enhancing dexterity in robotic manipulation via hierarchical contact exploration," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 390–397, 2023.

[69] S. Belkhale, Y. Cui, and D. Sadigh, "Data quality in imitation learning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[70] H. Zhu, T. Zhao, X. Ni, J. Wang, K. Fang, L. Righetti, and T. Pang, "Should we learn contact-rich manipulation policies from sampling-based planners?" *arXiv preprint arXiv:2412.09743*, 2024.

[71] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," in *Conference on Robot Learning*. PMLR, 2022, pp. 297–307.

[72] H. Qi, A. Kumar, R. Calandra, Y. Ma, and J. Malik, "In-hand object rotation via rapid motor adaptation," in *Conference on Robot Learning*. PMLR, 2023, pp. 1722–1732.

[73] M. Vecerik, T. Hester, J. Scholz, F. Wang, O. Pietquin, B. Piot, N. Heess, T. Rothörl, T. Lampe, and M. Riedmiller, "Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards," *arXiv preprint arXiv:1707.08817*, 2017.

[74] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 6292–6299.

[75] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning complex dexterous manipulation with deep reinforcement learning and demonstrations," *arXiv preprint arXiv:1709.10087*, 2017.

[76] H. Hu, S. Mirchandani, and D. Sadigh, "Imitation bootstrapped reinforcement learning," *arXiv preprint arXiv:2311.02198*, 2023.

[77] N. Hansen, Y. Lin, H. Su, X. Wang, V. Kumar, and A. Rajeswaran, "Modem: Accelerating visual model-based reinforcement learning with demonstrations," *arXiv preprint arXiv:2212.05698*, 2022.

[78] Y. Zhu, Z. Wang, J. Merel, A. Rusu, T. Erez, S. Cabi, S. Tunyasuvunakool, J. Kramár, R. Hadsell, N. de Freitas,

*et al.*, "Reinforcement and imitation learning for diverse visuomotor skills," *arXiv preprint arXiv:1802.09564*, 2018.

[79] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp: Adversarial motion priors for stylized physics-based character control," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 4, pp. 1–20, 2021.

[80] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions On Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[81] J.-P. Sleiman, M. Mittal, and M. Hutter, "Guided reinforcement learning for robust multi-contact loco-manipulation," in *8th Annual Conference on Robot Learning (CoRL 2024)*, 2024.

[82] J. Li, Y. Zhu, Y. Xie, Z. Jiang, M. Seo, G. Pavlakos, and Y. Zhu, "Okami: Teaching humanoid robots manipulation skills through single video imitation," in *8th Annual Conference on Robot Learning*, 2024.

[83] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar, "Mimicplay: Long-horizon imitation learning by watching human play," *arXiv preprint arXiv:2302.12422*, 2023.

[84] M. Seo, H. A. Park, S. Yuan, Y. Zhu, and L. Sentis, "Legato: Cross-embodiment imitation using a grasping tool," *arXiv preprint arXiv:2411.03682*, 2024.

[85] J. Yang, C. Glossop, A. Bhorkar, D. Shah, Q. Vuong, C. Finn, D. Sadigh, and S. Levine, "Pushing the limits of cross-embodiment learning for manipulation and navigation," *arXiv preprint arXiv:2402.19432*, 2024.

[86] R. Doshi, H. R. Walke, O. Mees, S. Dasari, and S. Levine, "Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation," in *8th Annual Conference on Robot Learning*.

[87] R. Tedrake and the Drake Development Team, "Drake: Model-based design and verification for robotics," 2019. [Online]. Available: https://drake.mit.edu

[88] G. Yang, "VUER: A 3d visualization and data collection environment for robot learning," 2024. [Online]. Available: https://github.com/vuer-ai/vuer

[89] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of operations research*, vol. 134, pp. 19–67, 2005.

[90] X. Li, T. Zhao, X. Zhu, J. Wang, T. Pang, and K. Fang, "Planning-guided diffusion policy learning for generalizable contact-rich bimanual manipulation," *arXiv preprint arXiv:2412.02676*, 2024.

| Parameter | $T$ | Plan Duration | $q_o$ | $q_r$ | $r_u$ |
|---|---|---|---|---|---|
| Floating Allegro Hand | 6 | 1.25 s | 10 | 0.01 | 0.1 |
| Bimanual iiwa Arms | 6 | 1.25 s | 10 | 0.01 | 10 |
| Bimanual Panda Arms | 6 | 2.0 s | 10 | 0.01 | 10 |

TABLE III: **Parameters for CEM.** $T$: planning horizon. $q_o$: scalar weight for tracking object trajectories. $q_r$: scalar weight for tracking robot trajectories. $r_u$: scalar weight for control input.

| Parameter | $T_o$ | $T_a$ | Freq | Epochs | Obs. Dim. | Act. Dim. |
|---|---|---|---|---|---|---|
| Floating Allegro Hand | 10 | 40 | 50 | 1000 | 34 | 22 |
| Bimanual iiwa Arms | 10 | 40 | 20 | 800 | 26 | 14 |
| Bimanual Panda Arms | 10 | 40 | 50 | 800 | 26 | 14 |

TABLE IV: **Parameters for diffusion policies.** $T_o$: observation horizon. $T_a$: action horizon. Freq: environment frequency (Hz, both observations and actions).

## IX. APPENDIX

In appendix, we present the implementation details of CEM and policy training.

### *A. CEM Implementation Details*

We provide detailed parameters for the CEM implementation in Table III. We optimize over the action knot points $u_{0:T-1}$, which are linearly interpolated to generate action commands sent to Drake. Drake simulates the contact dynamics $f$ at 200 Hz. The state cost matrix $Q_t = diag(q_o \cdot \mathbf{1}_{n_o}, q_r \cdot \mathbf{1}_{n_r})$, where $n_o$ and $n_r$ denote the object and robot state dimensions, and $\mathbf{1}$ is a vector or all 1's. The terminal state cost matrix $Q_T = 10 \cdot Q_t$. The input cost matrix $R_t = diag(r_u \cdot \mathbf{1}_{n_u})$, where $n_u$ represents the control input dimension. All of the systems adopt 50 samples, 5 elites and initial standard deviation $\sigma = 0.05 \cdot \mathbf{1}_{n_u}$ for action sampling.

### *B. Policy Implementation Details*

We train UNet-based diffusion policies [26] for all tasks. The action space is the robot configuration (joint angles, and additional floating base coordinates for the Allegro hand), while the observation space is the robot configuration and object pose (with orientations represented by rotation matrices). Detailed parameters are listed in Table IV.