

# Functional Force-Aware Retargeting from Virtual Human Demos to Soft Robot Policies

Uksang Yoo<sup>\*†</sup>, Mengjia Zhu<sup>†</sup>, Evan Pezent<sup>†</sup>, Jom Preechayasomboon<sup>†</sup>, Jean Oh<sup>\*</sup>, Jeffrey Ichnowski<sup>\*</sup>, Amir Memar<sup>†</sup>, Ben Abbatematteo<sup>†</sup>, Homanga Bharadhwaj<sup>†</sup>, Ashish Deshpande<sup>†</sup>, Harsha Prahlad<sup>†</sup>

<sup>\*</sup>Carnegie Mellon University, Robotics Institute

<sup>†</sup>Meta

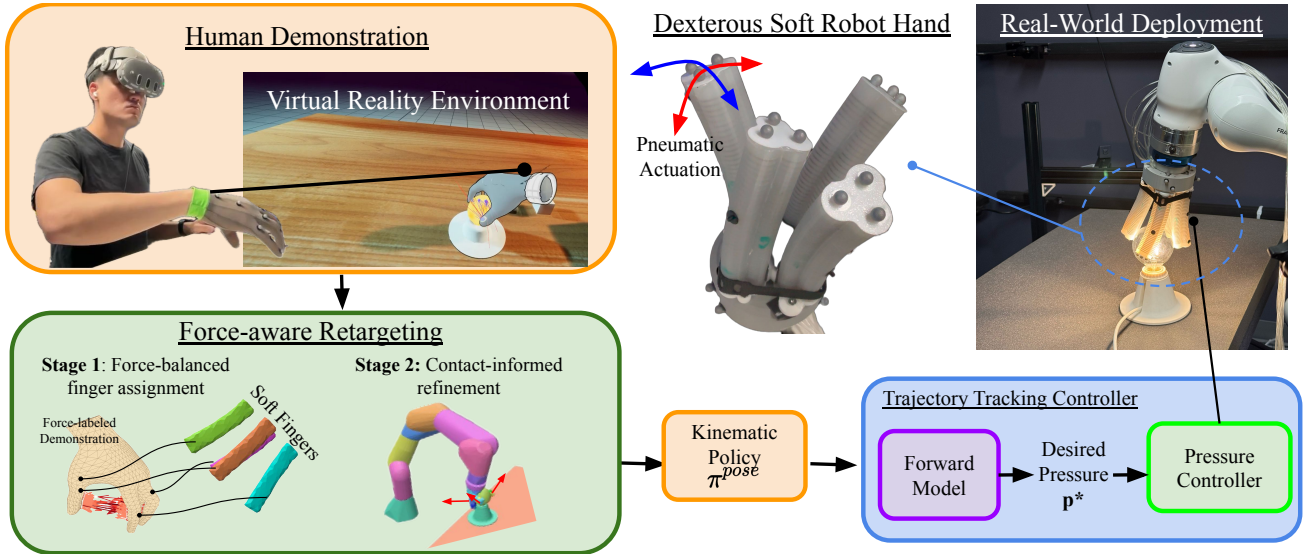


Fig. 1: **SoftAct** transfers human manipulation skills from VR demonstrations to a non-anthropomorphic pneumatic soft robot hand. The system uses contact force data to map human hand interactions onto the robot, enabling stable, functional manipulation across diverse tasks despite differences in hand structure and actuation.

**Abstract**—We introduce **SoftAct**, a framework for teaching soft robot hands to perform human-like manipulation skills by explicitly reasoning about contact forces. Leveraging immersive virtual reality, our system captures rich human demonstrations, including hand kinematics, object motion, dense contact patches, and detailed contact force information. Unlike conventional approaches that retarget human joint trajectories, **SoftAct** employs a two-stage, force-aware retargeting algorithm. The first stage attributes demonstrated contact forces to individual human fingers and allocates robot fingers proportionally, establishing a force-balanced mapping between human and robot hands. The second stage performs online retargeting by combining baseline end-effector pose tracking with geodesic-weighted contact refinements, using contact geometry and force magnitude to adjust robot fingertip targets in real time. This formulation enables soft robotic hands to reproduce the functional intent of human demonstrations while naturally accommodating extreme embodiment mismatch and nonlinear compliance. We evaluate **SoftAct** on a suite of contact-rich manipulation tasks using a custom non-anthropomorphic pneumatic soft robot hand. **SoftAct**'s controller reduces fingertip trajectory tracking RMSE by up to 55% and reduces tracking variance by up to 69% compared to kinematic and learning-based baselines. At the policy level, **SoftAct** achieves consistently higher success in zero-shot real-world deployment and in simulation. These results demonstrate that explicitly modeling contact geometry and force distribution is essential for effective skill transfer to

soft robotic hands, and cannot be recovered through kinematic imitation alone. Project videos and additional details are available at <https://soft-act.github.io>.

## I. INTRODUCTION

Soft robot hands offer intrinsic compliance, safety in human interaction, and robustness to uncertainty, making them particularly well suited for contact-rich manipulation tasks [32, 36, 37, 6, 38]. These advantages, however, come with fundamental challenges for policy learning and control [25]. In particular, translating dexterous manipulation skills from human demonstrations to soft robotic hands remains difficult due to extreme differences in workspaces, affordances, and contrability.

Unlike rigid and anthropomorphic robot hands, soft robotic hands are typically underactuated and driven with nonlinear actuators such as artificial muscles, exhibiting nonlinear, deformation-based motion rather than jointed kinematics [1, 29]. Their morphologies and finger configurations often differ substantially from the human hand in degrees of freedom, workspace geometry, and contact affordances [3]. As a result, direct correspondence between human and robot configurations is ill-defined, rendering classical imitation and retargeting approaches ineffective for many soft-hand systems.

Prior work on learning from demonstration and hand re-targeting often assumes a meaningful kinematic mapping and finger-to-finger correspondences between the demonstrator and the robot [27, 4]. Human joint trajectories, fingertip positions, or full hand poses are mapped to robot configurations via inverse kinematics, heuristic rules, or learned pose-to-pose mappings [7, 23, 24, 16]. While effective for anthropomorphic robot hands with similar structure, these methods break down for non-anthropomorphic and soft hands as we demonstrate in our experiments, where kinematic matching is often ill-posed and insufficient to reproduce functional manipulation, even when approximate geometric similarity is achieved.

Contact information can provide some guidance in generalizing manipulation strategies from human demonstrators to the robot embodiments [15, 18, 16]. However two limitations exist: obtaining such contact labeled datasets and these methods still require an explicit mapping between the hand surfaces to align the contact patches [15], limiting their applications to mostly anthropomorphic target embodiments.

In this paper, we propose `SoftAct`, a *contact force-guided* learning-from-demonstration framework for transferring human manipulation skills to non-anthropomorphic soft robotic hands. Our approach leverages immersive virtual reality to collect human demonstrations, capturing not only hand kinematics and object motion, but also dense contact patches and detailed contact force information throughout manipulation. Rather than directly re-targeting human joint trajectories, `SoftAct` introduces a two-stage force-aware re-targeting algorithm. In the first stage, contact forces are attributed to individual human fingers, and robot fingers are allocated proportionally, establishing a force-balanced mapping between the human and robot hands. In the second stage, online re-targeting is performed by combining baseline end-effector pose tracking with geodesic-weighted fine adjustments, using contact and force data to refine robot fingertip targets in real time. This formulation enables the robot to reproduce the functional demonstration by explicitly optimizing for geometry in contact points, while respecting the unique kinematic and morphological constraints of soft robotic hands. By optimizing contact- and force-based objectives under the constraints of a pneumatically actuated soft hand, `SoftAct` naturally discovers robot-specific manipulation strategies that may differ substantially from human motion, yet achieve similar interaction outcomes. This stands in contrast to kinematics-driven re-targeting methods, which enforce geometric similarity and often struggle under severe embodiment mismatch.

We evaluate our approach on multiple contact-rich manipulation tasks using a custom non-anthropomorphic pneumatic soft robot hand. Across tasks, our contact-centric re-targeting produces more stable interactions and higher task success rates than kinematics-only baselines. Ablation studies further highlight the contributions of individual system components. The resulting behaviors exhibit distinct contact patterns and force distributions that reflect the unique affordances of pneumatic actuation, demonstrating that contact-level representations are essential for transferring human manipulation skills to soft

robotic hands. Finally, we validate the simulated results with zero-shot real-world deployment, showing that `SoftAct` produces more stable and reliable rollout performances.

In summary, we make the following contributions:

- A contact-centric learning-from-demonstration formulation for re-targeting human manipulation skills to non-anthropomorphic soft robotic hands.
- A force-aware re-targeting method that explicitly optimizes contact patches and force distributions rather than hardcoded kinematic similarity.
- A state-of-the-art learning-based low-level controller for tracking trajectories with the highly nonlinear pneumatic soft robot hand.
- An end-to-end system for deploying the policy trained from virtual reality datasets on a pneumatically actuated soft hand, evaluated across multiple contact-rich tasks.

## II. RELATED WORK

We discuss prior works in dexterous manipulation with soft hands, learning from expert demonstrations, and human to robot motion re-targeting.

### A. Dexterous Soft Robot Manipulation

Soft robotic hands have been widely studied for manipulation due to their intrinsic compliance, which enables passive adaptation to object geometry, robustness to uncertainty, and safe interaction with humans and fragile objects [25, 6, 40]. By distributing contact forces over extended surface areas, soft hands can form stable grasps and tolerate positioning errors that would lead to failure in rigid, jointed hands [1, 29, 17].

Prior work has improved the dexterity of soft hands through advances in actuator design, material composition, and bio-inspired morphologies [22, 10]. These developments have enabled increasingly sophisticated grasping behaviors and limited forms of in-hand manipulation, often relying on carefully hand-designed primitives and substantial expert tuning [22]. Despite this progress, controlling soft hands remains fundamentally challenging due to their nonlinear, underactuated dynamics and deformation-driven motion, which complicate modeling, state estimation, and control [33].

As a result, most learning-based approaches for soft robotic manipulation focus on grasp acquisition or stabilization rather than sustained dexterous manipulation [13]. Progress is further constrained by limited proprioceptive sensing and the difficulty of collecting high-quality demonstrations for soft hands [30]. While the absence of rigid skeletal structures grants soft hands robustness and adaptability [20], it also introduces a severe mismatch between human hand kinematics and soft robot motion. Our work addresses this gap by moving beyond kinematic imitation and instead treating contact geometry and force as the primary transferable representations for dexterous manipulation.

### B. Learning from Expert Demonstrations for Manipulation

Learning from demonstration (LfD) provides a powerful paradigm for transferring complex manipulation skills from

humans to robots without explicit task modeling or reward specification [5]. Demonstrations have been collected using a range of modalities, including kinesthetic teaching [39], teleoperation [14, 2, 12], motion capture, and video-based observation. These demonstrations are typically used to supervise imitation learning or behavior cloning policies, requiring aligned observation–action pairs from expert executions [8, 31].

LfD has been explored as a means to reduce the substantial manual effort required to program soft robot behaviors [38, 19]. However, when there is significant embodiment mismatch between the human demonstrator and the robot, direct imitation of joint trajectories or hand poses becomes ill-posed. This challenge is especially pronounced for non-anthropomorphic and soft robotic hands, whose workspaces, degrees of freedom, and actuation mechanisms differ fundamentally from those of human hands. In such settings, geometric similarity alone is insufficient to capture the functional intent of the demonstrated manipulation.

### C. Human-to-Robot Motion Retargeting

Motion retargeting aims to map human demonstrations to robot actions, often serving as an initial step for learning-from-human pipelines. Common retargeting methods rely on kinematic correspondence between human and robot hands, mapping joint angles, fingertip positions, or full hand poses using inverse kinematics, heuristic rules, or learned pose-to-pose mappings [24, 15, 34]. These approaches have shown strong performance for anthropomorphic robot hands with similar kinematic structures, but their effectiveness degrades significantly for non-anthropomorphic or soft hands.

To relax strict kinematic correspondence, recent work has explored task-space retargeting, object-centric alignment, and contact-aware formulations [21, 15]. Some methods incorporate contact states or force-closure constraints to improve robustness, while others adopt optimization-based or policy-learning formulations to tolerate geometric mismatch [34, 18]. However, these prior works require explicit surface correspondence, and is primarily studied in the context of rigid, anthropomorphic hands [15, 16].

Retargeting remains particularly challenging for soft and underactuated hands due to compliance, limited controllability, and nonlinear actuation [29, 33]. Our work directly addresses these challenges by formulating retargeting as a contact- and force-centric optimization problem with simulation in the loop. By explicitly leveraging demonstrated contact patches and force distributions, our method enables soft robotic hands to reproduce the functional intent of human demonstrations while naturally exploiting their own morphology and actuation capabilities, rather than enforcing kinematic similarity.

## III. SOFTACT

Mapping human manipulation demonstrations to soft robotic hands is challenging due to severe embodiment mismatch. Differences in morphology, actuation, and controllabil-

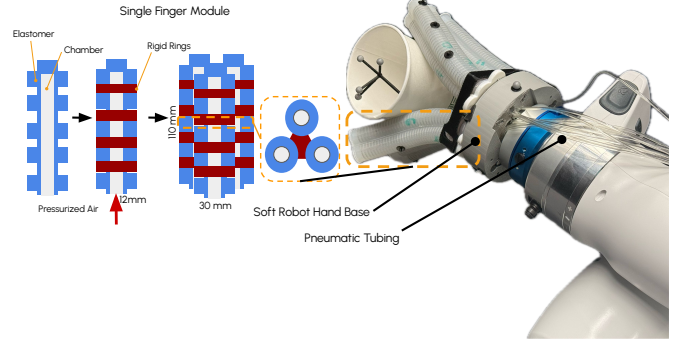


Fig. 2: **Design of the pneumatic soft manipulator.** The manipulator features a rigid–soft hybrid architecture, with each soft finger actuated by three radially arranged pneumatic chambers. The elastomeric finger body is reinforced with internal rigid structures to enhance controllability and repeatability. Differential pressurization enables planar and out-of-plane bending, while the fingers are mounted on a rigid base attached to a 7-DoF robotic arm.

ity cause naive kinematic retargeting to often fail to preserve the functional intent of manipulation.

We introduce *SOFTACT*, a force-aware retargeting framework that transfers human demonstrations to soft-fingered robotic hands by explicitly reasoning over contact geometry and force distribution. Rather than enforcing pose-level correspondence, *SOFTACT* treats contact forces as the primary transferable representation across embodiments.

Formally, given a human demonstration dataset

$$\mathcal{D}_h = \{(B_t, V_t, C_t)\}_{t=1}^T,$$

where  $C_t$  denotes hand–object contact locations and forces, our goal is to learn a robot control policy  $\pi(a_t | o_t)$  that achieves functionally equivalent object interactions.

*SOFTACT* achieves this through a two-stage retargeting process as shown in Fig 6. First, demonstrated contact forces are redistributed across robot fingers to produce a force-balanced human–robot mapping. Second, fingertip targets are refined online using geodesic-weighted contact influence on the hand surface, enabling stable, contact-consistent manipulation under extreme embodiment mismatch. The resulting policy is executed using a learned low-level controller for soft robotic actuation.

### A. Soft Robot Manipulator

Each soft finger (Fig. 2) is actuated by three parallel pneumatic chambers arranged radially around the finger axis. Inspired by prior work on reproducible soft robot design and fabrication using 3D-printed internal structures [11, 35], we adopt a rigid–soft hybrid architecture that combines compliant elastomeric materials with embedded rigid elements to improve controllability and repeatability.

The finger body is cast from Ecoflex 00–30, providing high compliance and large deformation under modest pressure inputs. Internally, each finger contains a rigid strain-limiting structures that constrain undesired deformation modes such as

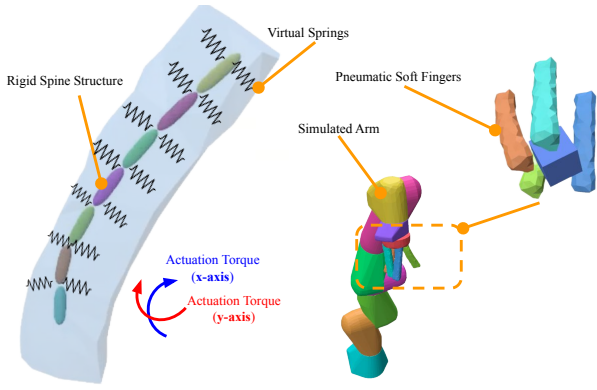


Fig. 3: **Simulation Setup.** We approximate pneumatic pressure-induced bending using internal torque-based actuation applied to a rigid spine, which deforms the surrounding soft finger through distributed virtual spring constraints.

ballooning and promote repeatable bending behavior. Differential pressurization of the three chambers produces planar bending and limited out-of-plane motion. The soft fingers are mounted on a rigid end-effector base attached to a 7-DoF robotic arm, allowing independent control of global end-effector motion and local finger deformation. The internal pressure to the soft robot chambers are controlled precisely at 200 Hz with a proportional pressure regulator.

### B. Simulation Environment

Retargeting and evaluation are performed in a custom finite-element-based (FEM) physics engine that supports coupled rigid and soft body dynamics. As shown in Fig. 3, the simulated platform consists of a 7-DoF Franka Emika Panda arm equipped with a multi-fingered soft robotic hand, enabling contact-rich dexterous manipulation under controlled and repeatable conditions.

The Franka arm is modeled as a rigid-body manipulator, while each soft finger is represented as a deformable FEM body with material parameters chosen to approximate the mechanical behavior of the physical elastomer. To enable controllable deformation, pneumatic actuation is approximated in simulation through applied torques that induce articulated bending of the fingers. This abstraction captures the dominant deformation and contact behaviors relevant to manipulation while maintaining computational efficiency.

Together, this hybrid rigid-soft simulation framework provides a stable and expressive testbed for evaluating force-aware retargeting and contact-centric manipulation policies.

### C. Human Demonstration Collection in Virtual Reality

Human manipulation demonstrations are collected in an immersive virtual reality (VR) environment built using Unreal Engine. Unlike conventional VR-based teleoperation systems [24], which use VR interfaces to directly control a physical robot in real time, our VR environment is used solely as a high-fidelity data collection platform for capturing rich human manipulation demonstrations.

The VR environment consists of a tabletop workspace that mirrors the geometry and scale of the real-world manipulation setup. Participants perform manipulation tasks using their right hand while interacting with virtual objects placed on the table.

Hand motion is captured using an OptiTrack motion capture system comprising ten cameras and 24 markers on the hand key points, providing high-fidelity tracking of the hand pose throughout each demonstration. The tracked hand pose is mapped in real time to a kinematic hand model in the VR environment, enabling accurate reproduction of hand motion and interaction geometry. Object poses are simultaneously recorded in a consistent world frame.

During each trial, we log full hand kinematics, object poses, and detailed interaction information at every timestep as shown in Fig. 4. Specifically, we extract contact patches between the hand and the object, represented as spatial distributions of contact points on the hand surface, along with the corresponding contact force magnitudes and directions. This rich contact-level information allows us to capture not only where contact occurs, but also how forces are applied and distributed over time. By collecting demonstrations in VR, we obtain precise and noise-free measurements of contact geometry and force that are difficult to acquire reliably in the real world. These demonstrations serve as the primary input to our force-aware retargeting pipeline, enabling the transfer of human manipulation strategies to a non-anthropomorphic soft robotic hand.

*a) Stage 1: Force-balanced finger assignment:* When transferring contact information across the hand, it is important that contact influence propagates along the physical surface of the hand rather than through free space. Using Euclidean distance in such cases would incorrectly couple physically unrelated regions of the hand.

To account for this, we measure distances intrinsically along the hand surface using mesh geodesics. Given two surface points  $x$  and  $y$ , their geodesic distance is computed over the hand mesh graph,

$$d_{\text{geo}}(x, y) = \min_{P \in \mathcal{P}(\hat{v}(x), \hat{v}(y))} \sum_{(i,j) \in P} \|v_i - v_j\|_2,$$

ensuring that contact influence respects surface curvature and articulated geometry, which is critical for modeling interactions on soft fingers. Full implementation details are provided in the appendix.

Due to embodiment mismatch between the human hand and the soft robotic hand, direct one-to-one finger correspondence is generally infeasible. Instead, we estimate the cumulative contact load associated with each human finger over the demonstration and allocate robot fingers proportionally, allowing many-to-one mappings.

Contact forces are diffused over the hand surface using a geodesic heat kernel. For mesh vertex  $v$  at timestep  $t$ , the heat value is defined as

$$h_v^t = \sum_{(c,f) \in \mathcal{C}_t} \|f\|_2 \exp(-\lambda d_{\text{geo}}(v, c)).$$

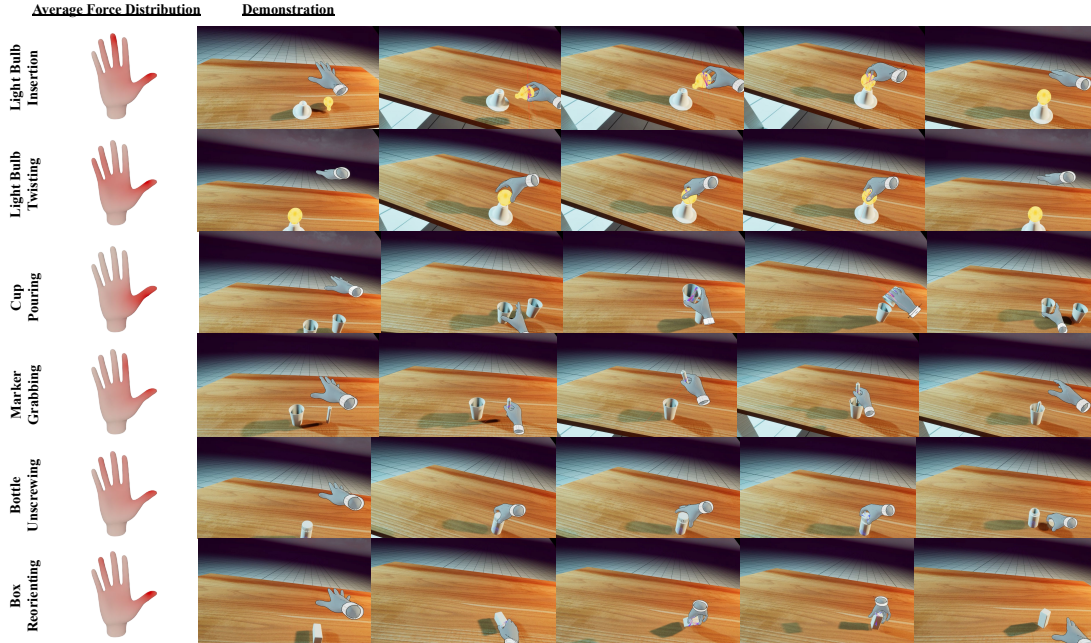


Fig. 4: **Contact-Rich Manipulation Tasks and Demonstration Force Profiles.** We evaluate `SoftAct` on six manipulation tasks (rows): *light bulb insertion*, *light bulb twisting*, *cup pouring*, *marker grabbing*, *bottle unscrewing*, and *box reorienting*. For each task, the left column visualizes the *average demonstrated contact-force distribution* over the human hand, highlighting task-dependent asymmetries (e.g., thumb-dominant vs. distributed contact). The remaining columns show representative time-lapse frames from a single VR demonstration for the task.

Heat values are accumulated over time and summed within each skeleton finger region to estimate the total load  $F_r$  attributed to finger  $r$ .

Robot fingers are then allocated to balance contact load by solving

$$\min_{\{n_r\}} \max_r \frac{F_r}{n_r} \quad \text{s.t.} \quad \sum_r n_r = N_f,$$

where  $n_r$  denotes the number of robot fingers assigned to skeleton finger  $r$ , and  $N_f$  is the total number of available robot fingers. The resulting assignment remains fixed throughout execution.

*b) Stage 2: Retargeting with contact-informed refinement:* During execution, the robot follows a baseline end-effector trajectory while each soft finger  $i \in \{1, \dots, N_f\}$  tracks its assigned human skeleton fingertip  $\pi(i)$ . To incorporate local contact interactions and align the contact surface of the human hand with the soft robot’s contact surface, we apply a contact-informed adjustment computed from nearby demonstrated contacts.

At timestep  $t$ , demonstrated contacts are given by  $\mathcal{C}_t = \{(c_j^t, f_j^t)\}_{j=1}^{|\mathcal{C}_t|}$ , where  $c_j^t \in \mathbb{R}^3$  denotes a contact location on the human hand surface and  $f_j^t \in \mathbb{R}^3$  is the corresponding contact force. Let  $s_i^t \in \mathbb{R}^3$  denote the current fingertip position of robot finger  $i$  expressed in the end-effector frame.

Each contact contributes a geodesic-weighted influence

$$w_{ij}^t = \|f_j^t\|_2 \exp(-\lambda d_{\text{geo}}(s_i^t, c_j^t)),$$

where  $d_{\text{geo}}(\cdot, \cdot)$  denotes the geodesic distance on the hand surface.

The fingertip target adjustment is computed as a weighted average

$$\delta_i^t = \frac{\sum_j w_{ij}^t (c_j^t - s_i^t)}{\sum_j w_{ij}^t + \epsilon}, \quad \|\delta_i^t\|_2 \leq \delta_{\text{max}}.$$

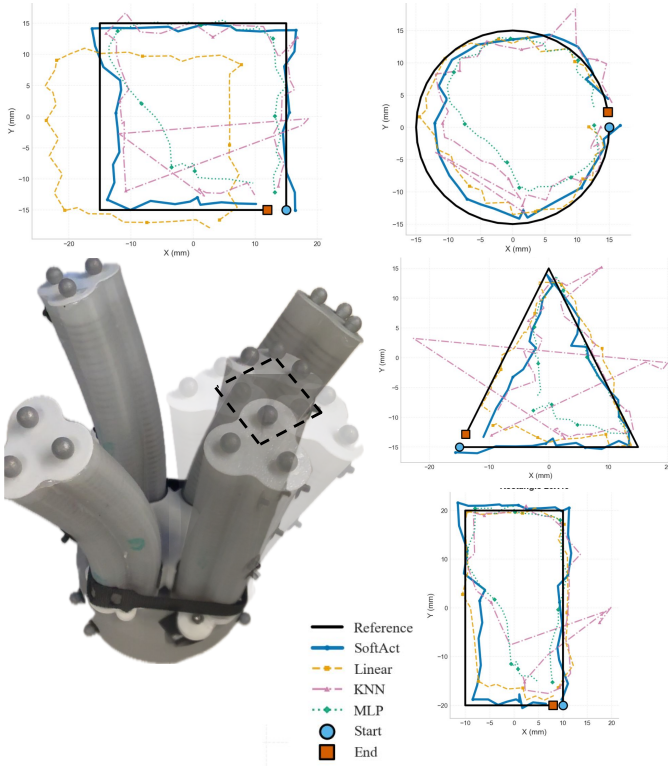
This refinement enables each robot finger to adapt its target online based on demonstrated contact geometry and force distribution, while remaining anchored to the assigned human fingertip. Applying this procedure across the demonstration produces a set of functionally retargeted trajectories  $\mathcal{D}_r$ , which are subsequently used for imitation learning.

#### D. Policy Learning

We train a diffusion policy [8] to imitate retargeted trajectories. The observation space is defined by robot proprioceptive state and object poses and the policy outputs a composite action consisting of an EE displacement, an EE orientation update, and soft-finger commands in torques as defined in Sec III-B. Proprioceptive and object features are encoded with MLPs and concatenated before being passed to the diffusion model backbone. The policy outputs denoised actions at a fixed control rate, with all inputs and outputs normalized and clamped to system limits. During real-world deployment, the object poses are obtained with reflective motion capture markers placed on the surface of the objects; we leave vision-based sim2real for future work.

#### E. Soft Finger Control

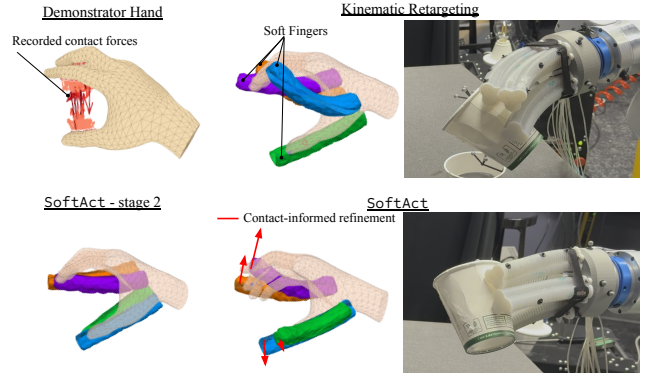
Given the desired fingertip trajectories produced by the policy, we require a low-level controller capable of accurately



**Fig. 5: Low-level Control Performance.** We evaluate trajectory tracking accuracy for a single soft finger executing planar reference trajectories, including square, circular, triangular, and rectangular motions. Each plot shows the desired fingertip trajectory in the  $xy$  plane (black curve) and the executed fingertip trajectories produced by different controllers. Colored curves correspond to controller rollouts. Tracking error is computed as the Euclidean distance between the executed position and the reference trajectory at each timestep. The proposed controller produces trajectories that closely follow the reference paths with low bias and variance, whereas baseline controllers exhibit noticeable drift, distortion, and accumulated error.

tracking these targets using the available actuation degrees of freedom. Controlling soft robot fingers is fundamentally challenging due to their nonlinear, underactuated, and deformation-driven dynamics. Unlike rigid hands with well-defined joint coordinates, pneumatic soft fingers exhibit complex pressure–deformation relationships that are difficult to model analytically and vary across the workspace. As a result, direct kinematic inversion or linear control is insufficient for precise fingertip positioning. To address this, we introduce a learned low-level controller that maps actuator commands to fingertip motion.

Each soft finger is actuated with three pneumatic chambers. Inspired by prior works in learning-based nonlinear system control in other domains such as locomotion [28], we learn a per-finger forward kinematic model using a MLP  $f_{\text{MLP}} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ , which maps chamber pressures  $\mathbf{p} \in [p_{\min}, p_{\max}]^3$  to fingertip displacement  $\mathbf{d} \in \mathbb{R}^3$  within safe pressure limits. This learned model captures the nonlinear compliant behavior



**Fig. 6: Retargeting Stages.** Overview of the two-stage force-aware retargeting pipeline. Stage 1 performs offline force-balanced finger assignment, allocating robot fingers based on demonstrated contact force distribution. Stage 2 performs contact-informed refinement, adjusting fingertip targets using geodesic-weighted contact to align the contact surfaces between the human hand mesh and the soft robot.

of the soft fingers and is used by the inverse controller to compute pressure commands that track desired fingertip targets during execution.

At runtime, desired fingertip displacements  $\mathbf{d}^{\text{ref}}$ , defined relative to a per-finger baseline configuration, are converted into pressure commands by inverting the learned forward model through optimization. For each waypoint, we solve

$$\mathbf{p}^{\text{ref}} = \arg \min_{\mathbf{p} \in [p_{\min}, p_{\max}]^3} \|\Pi_{xy}(f_{\text{MLP}}(\mathbf{p})) - \Pi_{xy}(\mathbf{d}^{\text{ref}})\|_2^2, \quad (1)$$

where  $\Pi_{xy}(\cdot)$  extracts the planar  $(x, y)$  components. Optimization is restricted to the planar subspace since all low-level tracking trajectories are planar, and out-of-plane motion is not independently controllable. The optimization problem in Eq. (1) is solved using gradient-based optimization through the differentiable MLP forward model.

To enforce actuator limits while preserving differentiability, pressures are re-parameterized using an unconstrained variable  $\mathbf{u} \in \mathbb{R}^3$ :

$$\mathbf{p}(\mathbf{u}) = p_{\min} + (p_{\max} - p_{\min}) \sigma(\mathbf{u}), \quad (2)$$

where  $\sigma(\cdot)$  denotes the element-wise sigmoid function. This transforms the constrained optimization in Eq. (1) into an unconstrained problem over  $\mathbf{u}$  while still allowing the optimizer to approach the full pressure range.

*a) Input and output normalization.*: During training, both pressure inputs and displacement outputs are standardized. At inference time, the same normalization is applied inside the optimization loop:

$$\tilde{\mathbf{p}} = (\mathbf{p} - \boldsymbol{\mu}_p) / \boldsymbol{\sigma}_p, \quad \tilde{\mathbf{d}} = (\mathbf{d} - \boldsymbol{\mu}_d) / \boldsymbol{\sigma}_d, \quad (3)$$

and the forward model is evaluated in normalized space before converting predictions back to metric units.

For trajectory execution, we generate a sequence of desired fingertip displacements  $\{\mathbf{d}_t^{\text{ref}}\}_{t=1}^T$ . To improve convergence



Fig. 7: **Real-World Manipulation Results.** Sequential frames showing zero-shot real-world execution of contact-rich manipulation tasks with `SoftAct`, including paper cup pouring, box reorientation, and light bulb screwing. The robot successfully maintains stable contact and executes task-relevant object motion despite severe embodiment mismatch between the human demonstrator and the non-anthropomorphic soft robot hand.

speed and stability, each optimization is warm-started from the previous solution. We solve the optimization using Adam for a fixed number of iterations and command the resulting pressures to the hardware at each waypoint.

#### IV. EXPERIMENTAL EVALUATION

We evaluate `SoftAct`'s force-aware retargeting and control approach through a series of simulation and real-world experiments, assessing low-level fingertip trajectory tracking, object-level trajectory tracking in simulation, and task success rates across a range of challenging, contact-rich manipulation tasks.

##### A. Low-Level Trajectory Tracking

The success of the policy rollout relies on accurate low-level controller performance to execute the predicted actions. To assess `SoftAct`'s controller performance, we first evaluate fingertip tracking accuracy on a single soft finger (Finger 0) executing planar square, circular, triangular, and rectangular trajectories. Each trajectory consists of 40 waypoints and is executed for three consecutive loops, with a dwell time of 0.5 s at each waypoint. Motion-capture feedback is used to obtain ground-truth fingertip positions for all trials.

We compare four controllers: (i) `SoftAct`'s MLP-based inverse model optimized via gradient descent, (ii) a linear Jacobian controller derived from local kinematic linearization inspired by Sieler and Brock [26], (iii)  $k$ -nearest-neighbor regression over sampled pressure–displacement pairs, and (iv) direct MLP prediction of chamber pressures from desired fingertip displacements.

Table I reports trajectory tracking errors measured in the  $xy$  plane and Fig. 5 visualizes the resulting trajectories. The learned `SoftAct` low-level soft robot pressure controller consistently achieves lower tracking error and reduced variance compared to all baselines. Relative to the strongest baseline, `SoftAct` reduces RMSE by 55% and reduces mean tracking error by 48%. Compared to learning-based baselines, the proposed controller reduces RMSE by 63% and 74%,

TABLE I: Fingertip trajectory tracking performance for all controllers. Lower is better.

Controller	RMSE (mm)	Mean $\pm$ Std (mm)	Max (mm)
Direct KNN	8.74	5.33 $\pm$ 6.93	34.34
Direct MLP	6.11	4.78 $\pm$ 3.80	13.85
Direct Linear	5.05	3.97 $\pm$ 3.13	10.89
<code>SoftAct</code>	<b>2.28</b>	<b>2.06 <math>\pm</math> 0.98</b>	<b>5.47</b>

respectively, and substantially decreases worst-case tracking error. These results indicate that optimizing a learned inverse model at execution time yields more accurate and stable trajectory tracking than direct prediction methods. The improved precision and consistency support the use of `SoftAct` as a low-level execution module for contact-sensitive manipulation under nonlinear pneumatic actuation.

##### B. Object Trajectory Tracking in Simulation

We next evaluate how improvements in low-level fingertip tracking translate to object-level manipulation performance in simulation. Executed object trajectories are compared against ground-truth demonstrations for paper cup pouring, light bulb insertion, and light bulb screwing. We compare three retargeting conditions: a kinematic-only baseline, `SoftAct`-stage 1 with force-balanced finger assignment only, and the full `SoftAct` method with contact-informed refinement.

Table II reports translational and rotational deviations across tasks. Across all evaluated settings, `SoftAct` consistently outperforms a kinematic-only baseline, reducing translational error by approximately 55–75% and rotational error by up to 80%. The largest gains occur in constrained insertion and rotational tasks, where maintaining stable contact is critical and kinematic retargeting accumulates significant drift.

While `SoftAct` with just stage 1 improves performance relative to pure kinematic retargeting, it remains consistently inferior to the full contact-aware formulation. This gap indicates that post-hoc adaptation alone cannot recover the

TABLE II: Object trajectory tracking error in simulation experiments. Lower is better.

Task	Method	Pos. (m)	Rot. (deg)
Light Bulb Insertion	Kinematic	$1.52 \pm 0.61$	$11.8 \pm 3.4$
	SoftAct-stage 2	$0.97 \pm 0.42$	$25.1 \pm 7.9$
	SoftAct	<b><math>0.48 \pm 0.21</math></b>	<b><math>12.4 \pm 8.1</math></b>
Light Bulb Twisting	Kinematic	$0.05 \pm 0.07$	$18.2 \pm 6.5$
	SoftAct-stage 2	$0.07 \pm 0.03$	$9.4 \pm 3.4$
	SoftAct	<b><math>0.02 \pm 0.005</math></b>	<b><math>2.9 \pm 1.3</math></b>
Cup Pouring	Kinematic	$1.29 \pm 0.50$	$12.5 \pm 4.1$
	SoftAct-stage 2	$0.81 \pm 0.35$	$6.4 \pm 2.2$
	SoftAct	<b><math>0.51 \pm 0.21</math></b>	<b><math>3.5 \pm 1.3</math></b>
Marker Grasping	Kinematic	$0.97 \pm 0.41$	$14.3 \pm 5.0$
	SoftAct-stage 2	$0.56 \pm 0.25$	$7.3 \pm 2.5$
	SoftAct	<b><math>0.31 \pm 0.13</math></b>	<b><math>3.9 \pm 1.5</math></b>
Bottle Unscrewing	Kinematic	$0.26 \pm 0.11$	$21.3 \pm 7.2$
	SoftAct-stage 2	$0.11 \pm 0.05$	$11.1 \pm 3.8$
	SoftAct	<b><math>0.07 \pm 0.12</math></b>	<b><math>6.7 \pm 2.6</math></b>
Box Reorienting	Kinematic	$1.34 \pm 0.54$	$17.6 \pm 6.0$
	SoftAct-stage 2	$0.89 \pm 0.37$	$9.0 \pm 3.1$
	SoftAct	<b><math>0.62 \pm 0.26</math></b>	<b><math>5.1 \pm 2.0</math></b>

benefits of explicitly reasoning about contact geometry and force distribution during retargeting.

### C. Policy Evaluation

We evaluate policy-level performance on a suite of contact-rich manipulation tasks as listed in Fig. 4 in both simulation and the real world. These tasks span asymmetric and distributed contact regimes, requiring stable force regulation, coordinated multi-finger interaction, and robustness to severe embodiment mismatch.

In simulation, SoftAct achieves consistent and substantial improvements over a kinematic-only baseline (Table III), increasing task success rates by roughly 30–70% across manipulation settings. Gains are most pronounced for insertion and reorientation tasks, where kinematic policies frequently fail due to unstable or poorly distributed contact.

We further evaluate zero-shot real-world deployment, as shown in Fig. 7. SoftAct consistently outperforms the kinematic baseline across all evaluated tasks, achieving notable relative gains in real-world success rates. This suggests that explicitly reasoning about contact geometry and force distribution improves sim-to-real transfer for soft robotic manipulation.

Overall, the consistent gains in both simulation and real-world experiments indicate that explicitly modeling contact geometry and force distribution during retargeting is essential for reliable manipulation with non-anthropomorphic soft robotic hands, and cannot be recovered through kinematic imitation alone.

## V. CONCLUSION

In this paper, we presented a force-aware, contact-centric retargeting framework for transferring human manipulation

TABLE III: Task success rates in simulation and the real world for SoftAct compared to a kinematic-only baseline. For each baseline per task, we do 30 rollouts in simulation, and 20 in the real world. Higher is better.

Task	Domain	SoftAct	Kinematic Baseline
Paper Cup Pouring	Simulation	90%	40%
Light Bulb Insertion	Simulation	47%	0%
Marker Grasping	Simulation	97%	73%
Bottle Unscrewing	Simulation	80%	47%
Box Reorienting	Simulation	90%	30%
Light Bulb Screwing	Simulation	100%	77%
Paper Cup Pouring	Real	85%	35%
Light Bulb Screwing	Real	95%	30%
Box Reorienting	Real	70%	10%

skills to non-anthropomorphic soft robotic hands. Rather than enforcing kinematic correspondence between human and robot hands, our approach treats contact geometry and force as the primary transferable representations. By combining off-line force-balanced finger assignment with online geodesic-weighted contact refinements, the method enables soft robotic hands to reproduce the functional intent of human demonstrations while naturally exploiting their own morphology and compliance.

We demonstrated the effectiveness of our approach across multiple contact-rich manipulation tasks, including paper cup pouring, light bulb insertion, light bulb twisting, bottle unscrewing, box reorienting and marker grasping. Quantitative results show that force-aware retargeting significantly outperforms kinematic-only baselines in both tracking accuracy and task success. Overall, this work suggests that contact- and force-centric representations provide a promising pathway for bridging extreme embodiment mismatch in imitation learning, enabling dexterous manipulation with soft robotic hands beyond the limits of kinematic correspondence.

## VI. LIMITATIONS AND FUTURE WORK

Despite these results, several limitations remain. First, our current framework relies on access to ground-truth contact forces during demonstration, which can only be obtained in simulation. With recent progress in visual hand-object interaction understanding, extending the approach to operate with visually estimated contact forces would significantly broaden its applicability to real-world data collection. Additionally, our evaluation focuses on relatively short-horizon manipulation tasks with fixed interaction structure. Scaling the framework to longer-horizon, multi-stage manipulation remains an open challenge, particularly in settings where contact modes and functional roles evolve over time. Finally, for real-world deployment, we rely on object poses obtained from motion capture markers, which may not be available in many practical applications. An important direction for future work is to leverage the trained state-based policy as a teacher to train a vision-based student policy, enabling sim-to-real transfer without reliance on motion-capture markers.

## REFERENCES

- [1] Sylvain Abondance, Clark B Teeple, and Robert J Wood. A dexterous soft robotic hand for delicate in-hand manipulation. *IEEE Robotics and Automation Letters*, 5(4): 5502–5509, 2020.
- [2] Jorge Aldaco, Travis Armstrong, Robert Baruch, Jeff Bingham, Sanky Chan, Kenneth Draper, Debidatta Dwibedi, Chelsea Finn, Pete Florence, Spencer Goodrich, et al. Aloha 2: An enhanced low-cost hardware for bimanual teleoperation. *arXiv preprint arXiv:2405.02292*, 2024.
- [3] Gursel Alici. Softer is harder: What differentiates soft robotics from hard robotics? *MRS Advances*, 3(28):1557–1568, 2018.
- [4] Dafni Antotsiou, Guillermo Garcia-Hernando, and Tae-Kyun Kim. Task-oriented hand motion retargeting for dexterous manipulation imitation. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018.
- [5] Aude Billard and Danica Kragic. Trends and challenges in robot manipulation. *Science*, 364(6446):eaat8414, 2019.
- [6] Xiaoqian Chen, Xiang Zhang, Yiyong Huang, Lu Cao, and Jinguo Liu. A review of soft manipulator research, applications, and opportunities. *Journal of Field Robotics*, 39(3):281–311, 2022.
- [7] Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-television: Teleoperation with immersive active visual feedback. *arXiv preprint arXiv:2407.01512*, 2024.
- [8] 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. *The International Journal of Robotics Research*, page 02783649241273668, 2023.
- [9] Liang Ding, Lizhou Niu, Yang Su, Huaiguang Yang, Guangjun Liu, Haibo Gao, and Zongquan Deng. Dynamic finite element modeling and simulation of soft robots. *Chinese journal of mechanical engineering*, 35(1):24, 2022.
- [10] Charlotte Firth, Kate Dunn, M Hank Haeusler, and Yi Sun. Anthropomorphic soft robotic end-effector for use with collaborative robots in the construction industry. *Automation in Construction*, 138:104218, 2022.
- [11] J. Fras, J. Glowka, and K. Althoefer. Instant soft robot: A simple recipe for quick and easy manufacturing. In *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 482–488, 2020. doi: 10.1109/RoboSoft48309.2020.9115973.
- [12] Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.
- [13] Abhishek Gupta, Clemens Eppner, Sergey Levine, and Pieter Abbeel. Learning dexterous manipulation for a soft robotic hand from human demonstrations. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3786–3793. IEEE, 2016.
- [14] Ankur Handa, Karl Van Wyk, Wei Yang, Jacky Liang, Yu-Wei Chao, Qian Wan, Stan Birchfield, Nathan Ratliff, and Dieter Fox. Dexipilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9164–9170. IEEE, 2020.
- [15] Arjun S Lakshmipathy, Jessica K Hodgins, and Nancy S Pollard. Kinematic motion retargeting for contact-rich anthropomorphic manipulations. *ACM Transactions on Graphics*, 44(2):1–20, 2025.
- [16] Kailin Li, Puhao Li, Tengyu Liu, Yuyang Li, and Siyuan Huang. Maniptrans: Efficient dexterous bimanual manipulation transfer via residual learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6991–7003, 2025.
- [17] Fukang Liu, Kavya Puthuveetil, Akhil Padmanabha, Karan Khokar, Zeynep Temel, and Zackory Erickson. Skingrip: An adaptive soft robotic manipulator with capacitive sensing for whole-limb bathing assistance. *arXiv preprint arXiv:2405.02772*, 2024.
- [18] Zhao Mandi, Yifan Hou, Dieter Fox, Yashraj Narang, Ajay Mandlekar, and Shuran Song. Dexmachina: Functional retargeting for bimanual dexterous manipulation. *arXiv preprint arXiv:2505.24853*, 2025.
- [19] Muhammad Sunny Nazeer, Cecilia Laschi, and Egidio Falotico. Soft dagger: Sample-efficient imitation learning for control of soft robots. *Sensors*, 23(19):8278, 2023.
- [20] Amir Pagoli, Frédéric Chapelle, Juan Antonio Corrales, Youcef Mezouar, and Yuri Lapusta. A soft robotic gripper with an active palm and reconfigurable fingers for fully dexterous in-hand manipulation. *IEEE Robotics and Automation Letters*, 6(4):7706–7713, 2021.
- [21] Nancy S Pollard and Jessica K Hodgins. Generalizing demonstrated manipulation tasks. In *Algorithmic Foundations of Robotics V*, pages 523–539. Springer, 2004.
- [22] Steffen Puhmann, Jason Harris, and Oliver Brock. Rbo hand 3: A platform for soft dexterous manipulation. *IEEE Transactions on Robotics*, 38(6):3434–3449, 2022.
- [23] Yuzhe Qin, Hao Su, and Xiaolong Wang. From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation. *IEEE Robotics and Automation Letters*, 7(4):10873–10881, 2022.
- [24] Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. *arXiv preprint arXiv:2307.04577*, 2023.
- [25] Daniela Rus and Michael T Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475, 2015.
- [26] Adrian Sieler and Oliver Brock. Dexterous soft hands linearize feedback-control for in-hand manipulation. In

- 2023 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8757–8764. IEEE, 2023.
- [27] Aravind Sivakumar, Kenneth Shaw, and Deepak Pathak. Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube. *arXiv preprint arXiv:2202.10448*, 2022.
- [28] Bhavya Sukhija, Nathanael Köhler, Miguel Zamora, Simon Zimmermann, Sebastian Curi, Andreas Krause, and Stelian Coros. Gradient-based trajectory optimization with learned dynamics. *arXiv preprint arXiv:2204.04558*, 2022.
- [29] Clark B Teeple, Theodore N Koutros, Moritz A Graule, and Robert J Wood. Multi-segment soft robotic fingers enable robust precision grasping. *The International Journal of Robotics Research*, 39(14):1647–1667, 2020.
- [30] Abraham Itzhak Weinberg, Alon Shirizly, Osher Azulay, and Avishai Sintov. Survey of learning-based approaches for robotic in-hand manipulation. *Frontiers in Robotics and AI*, 11:1455431, 2024.
- [31] Mengda Xu, Han Zhang, Yifan Hou, Zhenjia Xu, Linxi Fan, Manuela Veloso, and Shuran Song. Dexumi: Using human hand as the universal manipulation interface for dexterous manipulation. *arXiv preprint arXiv:2505.21864*, 2025.
- [32] Yunchao Yao, Uksang Yoo, Jean Oh, Christopher G Atkeson, and Jeffrey Ichnowski. Soft robotic dynamic in-hand pen spinning. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7. IEEE, 2025.
- [33] Oncay Yasa, Yasunori Toshimitsu, Mike Y Michelis, Lewis S Jones, Miriam Filippi, Thomas Buchner, and Robert K Katzschmann. An overview of soft robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 6(1):1–29, 2023.
- [34] Zhao-Heng Yin, Changhao Wang, Luis Pineda, Krishna Bodduluri, Tingfan Wu, Pieter Abbeel, and Mustafa Mukadam. Geometric retargeting: A principled, ultra-fast neural hand retargeting algorithm. *arXiv preprint arXiv:2503.07541*, 2025.
- [35] Uksang Yoo, Yang Liu, Ashish D Deshpande, and Farshid Alamabeigi. Analytical design of a pneumatic elastomer robot with deterministically adjusted stiffness. *IEEE robotics and automation letters*, 6(4):7773–7780, 2021.
- [36] Uksang Yoo, Nathaniel Dennler, Maja Mataric, Stefanos Nikolaidis, Jean Oh, and Jeffrey Ichnowski. Moe-hair: Toward soft and compliant contact-rich hair manipulation and care. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 1163–1167, 2024.
- [37] Uksang Yoo, Nathaniel Dennler, Eliot Xing, Maja Mataric, Stefanos Nikolaidis, Jeffrey Ichnowski, and Jean Oh. Soft and compliant contact-rich hair manipulation and care. In *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 610–619. IEEE, 2025.
- [38] Uksang Yoo, Jonathan Francis, Jean Oh, and Jeffrey Ichnowski. Kinesoft: Learning proprioceptive manipulation policies with soft robot hands. In *Conference on robot learning*. PMLR, 2025.
- [39] Di Zhang, Chengbo Yuan, Chuan Wen, Hai Zhang, Junqiao Zhao, and Yang Gao. Kinedex: Learning tactile-informed visuomotor policies via kinesthetic teaching for dexterous manipulation. *arXiv preprint arXiv:2505.01974*, 2025.
- [40] Xiaolong Zhou, Xiaoting Chen, and Tianmiao Yang. Soft robotic grippers. *Advanced Intelligent Systems*, 5(1): 2000198, 2023.