KineSoft: Learning Proprioceptive Manipulation Policies with Soft Robot Hands

Uksang Yoo[1], **Jonathan Francis**[1,2], **Jean Oh**[1], **Jeffrey Ichnowski**[1]
[1]Carnegie Mellon University [2]Bosch Center for AI
{uksang, jmfrancis, jeanoh, jeffi}@cmu.edu

Abstract: Underactuated soft robot hands offer inherent safety and adaptability advantages over rigid systems. While imitation learning shows promise for acquiring complex dexterous manipulation skills, adapting existing methods to soft robots presents unique challenges in state representation and data collection. We propose KineSoft, a framework for direct kinesthetic teaching of soft robotic hands that leverages their natural compliance as a skill teaching advantage rather than only as a control challenge. With KineSoft, human demonstrators physically guide the robot while the system learns to associate proprioceptive patterns with successful manipulation strategies. KineSoft makes three key contributions: (1) a shape-based imitation learning framework that uses proprioceptive feedback to ground diffusion-based policies (2) a low-level shape-conditioned controller that enables precise tracking of desired shape trajectories, (3) a sim-to-real learning approach to soft robot mesh shape sensing with an internal strain-sensing array. In physical experiments, we demonstrate the superiority of KineSoft over the strain-based policy baseline in six in-hand manipulation tasks involving both rigid and deformable objects. KineSoft's results suggest that embracing the inherent properties of soft robots leads to intuitive and robust dexterous manipulation capabilities. More details and videos can be found in kinesoft-policy.github.io.

Keywords: Soft Robots, Proprioceptive Estimation, Imitation Learning

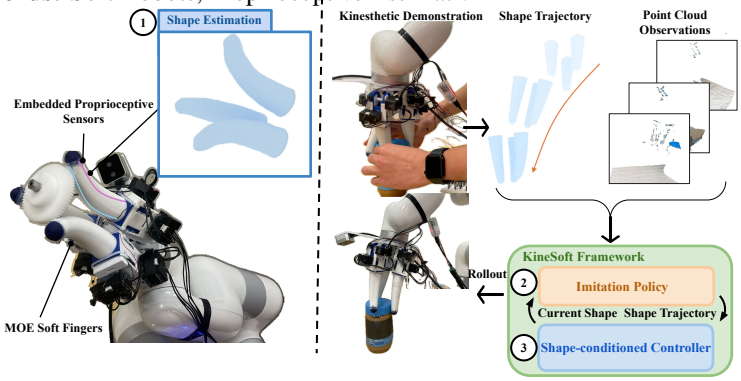

Figure 1: **KineSoft** is a framework for learning from kinesthetic demonstration, enabling free-shaped soft end-effectors to perform dexterous manipulation. Three key components are: 1) a proprioceptive model for high-fidelity shape estimation, 2) diffusion-based imitation learning for predicting the changes in shape and end-effector poses, and 3) a shape-conditioned controller that allows the soft hand to track given shape trajectories.

1 Introduction

Underactuated soft robot hands offer key advantages over rigid counterparts, including inherent safety through material compliance [1, 2, 3, 4] and robust adaptability to uncertain object geometries [5, 6, 7, 3]. These properties make them well-suited for applications requiring contact-rich

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

and reliable interactions such as assistive care, delicate object manipulation, fruit picking, and collaborative manufacturing [1, 8]. However, imparting dexterous in-hand manipulation skills to soft hands remains challenging. Existing methods for soft robot manipulation often rely on hand-crafted primitives [5, 7, 9, 10] that necessitate expert operators and limit system adaptability.

Recent advances in imitation learning, particularly frameworks like diffusion policy, have shown promise in teaching complex manipulation skills [11, 12, 13]. These approaches have been successfully applied to various tasks, from long-horizon mobile manipulation with rigid grippers [14] to deformable object manipulation with simple end-effectors [15]. Unlike reinforcement learning methods that depend on carefully crafted reward functions and simulation environments [16, 17], imitation learning requires only demonstration trajectories of successful task execution. However, collecting such demonstrations for soft robots presents challenges: demonstration collection methods for rigid articulated robots generally do not apply to underactuated soft robots and their virtually infinite degrees of freedom. Teleoperation interfaces [18] designed for rigid anthropomorphic hands fail to capture the unique capabilities and constraints of underactuated soft end-effectors. Soft robots often lack a direct mapping to rigid human hand joints. Additionally, standard robot state representations for rigid robots in imitation learning frameworks [19], such as rigid transformation poses, are ill-suited to provide meaningful state information for continuously deforming structures. Despite recent advances in expressive state representation learning for soft robots [20, 21, 22], these have not yet been applied to skill learning frameworks for in-hand manipulation. These limitations have restricted the application of imitation learning to soft robotic manipulation.

We propose KineSoft, a hierarchical framework to enable direct kinesthetic teaching of soft robot hands skills. The key insight is that soft robots' natural compliance provides an advantage for teaching rather than just a control challenge. Unlike rigid robots, whose joints and linkages often resist human guidance due to strict kinematic constraints, soft hands can be easily deformed by a human demonstrator into desired poses. This allows demonstrations to be collected by physically manipulating the hand without violating mechanical limits or producing infeasible configurations. KineSoft has three key components (Figure 1). (1) The proprioceptive system achieves state-of-the-art shape estimation using internal strain-sensing arrays and a model trained on large simulated data of the robot's high-dimensional configurations. These sensors provide rich proprioceptive feedback while preserving the hand's natural compliance, allowing KineSoft to capture detailed information about the hand's deformation state during manipulation tasks in real time. (2) We train an imitation policy on these shape trajectories and use it to generate deformation trajectories during rollout. (3) KineSoft's low-level shape-conditioned controller then tracks these desired shapes. Experiments demonstrate that KineSoft achieves accurate shape state estimation, mesh-based trajectory tracking through the shape-conditioned controller, and high performance in learned manipulation skills through these shape-based representations.

In summary, this paper contributes: **i)** KineSoft, a framework for learning from kinesthetic demonstrations for soft robot hands that enables dexterous in-hand manipulation, **ii)** a state-of-the-art proprioceptive shape estimation approach using strain sensing integrated with soft robot hands that enables precise tracking of finger deformations during contact-rich tasks, **iii)** shape-conditioned controller for tracking the generated deformation trajectories and performing dexterous manipulation tasks, and **iv)** simulated dataset and trained model for state estimation and control, which we demonstrate can be readily deployed to open-source soft robot hands, such as the MOE platform [1].

2 Related Work

Learning for in-hand dexterity. Recent advances in reinforcement learning have driven significant progress in rigid robot in-hand dexterity [17, 16, 7, 23]. However, these approaches face challenges in real-world deployment due to difficulty in creating resettable training environments. To mitigate these issues, many methods transfer policies trained in simulation to the real world, which has shown success with rigid robots with fine tuning in the real world [16, 17] but remains less applicable for soft robots due to the complexities involved in modeling deformable materials,

forward kinematics, and contact dynamics [24]. Despite recent interest in leveraging reinforcement learning for soft robot arm control and trajectory tracking [25, 26, 27], difficulty in modeling simultaneous contact and soft robot body deformation dynamics have hindered their application to soft robot dexterous manipulation.

Imitation learning has emerged as a promising alternative for reducing the reliance on explicit physics simulation environments, enabling robots to acquire manipulation skills efficiently with real-world data [28, 11, 19]. In-hand manipulation skills through imitation learning are typically achievable with anthropomorphic hands that provide a direct mapping between human hand and robot poses [29, 30, 31]. However, despite the inherent benefits in safety and dexterity through compliance of soft robot hands, they face unique challenges due to the absence of reliable proprioceptive feedback [32] and thus there is a lack of practical frameworks for collecting demonstrations for soft robots. To the best of our knowledge, *KineSoft* is the first framework that effectively leverages passive compliance of soft robots to collect demonstrations and enables soft robots to acquire dexterous in-hand manipulation skills.

Soft robot dexterity. Soft robot hands excel in grasping and manipulation tasks due to their material compliance, allowing for passive adaptation to diverse object geometries [33, 34]. This compliance facilitates robust grasping and safe interactions with humans and delicate objects by distributing contact forces [9, 35, 1]. Recent advances in soft robots have aimed to improve dexterity through innovative actuator designs, material improvements, and bioinspired morphologies [36, 8]. Despite these strides, controlling soft hands remains a significant challenge due to their complex dynamics and the high dimensionality of their state spaces [37]. Consequently, learning-based approaches for soft robot dexterity have primarily focused on grasping [38], while the development of dexterous in-hand manipulation skills has been hindered by the lack of available demonstration methods and reliable proprioceptive feedback [32]. The unique advantages of soft robot hands in robust in-hand manipulation stem from their lack of rigid skeletal structures [39]. However, this absence also introduces challenges, as soft robot kinematics differ significantly from human hand motions. The *KineSoft* framework bridges these gaps by integrating novel and accurate shape estimation methods with learned imitation policies, enabling efficient skill acquisition for dexterous manipulation with soft robots.

Soft robot state representation. Proprioceptive shape sensing is critical for enabling robust control in soft robots, particularly for accurate shape tracking and feedback-driven control [40, 41]. Existing works often use low degree-of-freedom shape representations such as constant curvature models [42, 43] or bending angles [44], which fail to capture the full richness of soft robot deformation states. Toward capturing these complex deformation behaviors of soft robot manipulators, recent approaches based on mechanics models have employed Cosserat rod models and high-dimensional Frenet-Serret frames, associated with the continuum cross sections [45]. However, updating and preserving hard constraints using these approaches is computationally expensive [45, 46]. Recent learning-based models have introduced more expressive representations of soft robot states using point clouds [20, 21] and meshes [22, 47]. However, these learned representations have not been connected to policy learning for dexterous manipulation tasks. Addressing this gap, *KineSoft* proposes a novel framework that leverages proprioceptive sensing and learned shape representations, based on vertex displacement fields over meshes, to facilitate dexterous manipulation skill learning in soft robots.

3 Problem Statement

We consider the problem of learning dexterous manipulation skills with a soft robot hand equipped with internal proprioceptive sensing. Let $\mathbf{M}$ denote the true continuous deformation state of the hand, which is governed by the robot's mechanical structure and actuated through control inputs $\mathbf{U}$. Since $\mathbf{M}$ is unobservable and high-dimensional [48] in practice, we assume access to an internal

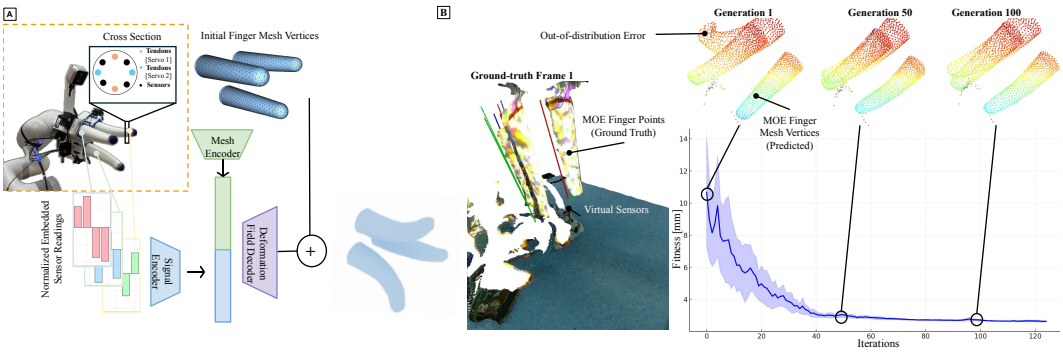

Figure 2: **Proprioception network.** A: Network architecture for mesh shape estimation of the soft fingers. B: Results of domain alignment iterations where the loss converged after 200 iterations.

sensor measurement space $\mathbf{S}$ (e.g., from strain sensors) that provides indirect information about the deformation.

The objective is to learn a policy $\pi : \mathbf{M} \times \mathcal{O} \rightarrow \mathbf{U}$ that maps from the current deformation state and additional task-relevant observations $\mathcal{O}$ (e.g., visual input) to a control action in $\mathbf{U}$. This setting poses three key challenges: i) Estimating the unobservable deformation state $\mathbf{M}$ from the sensor measurement space $\mathbf{S}$; ii) Learning a reliable mapping from $\mathbf{S}$ to $\mathbf{M}$ for downstream use in control; iii) Learning a control policy π that uses $\mathbf{M}$ and $\mathcal{O}$ to generate actions, where $\mathcal{O}$ may include exteroceptive modalities (e.g., vision) that are complementary to the proprioceptive signals in $\mathbf{S}$.

This presents unique challenges for soft robots compared to rigid systems: the continuous deformation space $\mathbf{M}$ is theoretically infinite-dimensional as a continuum, demonstrations must account for the robot's inherent compliance, and the mapping between actuation and deformation is nonlinear or difficult to simulate in its entirety [45]. The objective is to develop a framework that can effectively learn and execute manipulation skills while embracing these fundamental characteristics.

4 Method

KineSoft enables soft robot dexterous manipulation through a hierarchical approach that bridges the gap between kinesthetic teaching and autonomous execution. Our framework comprises five integrated components: a sensorized MOE-based soft hand with embedded strain sensors, a high-fidelity mesh-based shape estimation model, a shape-conditioned controller for precise trajectory tracking, a diffusion-based shape imitation policy, and a low-level controller that tracks desired shapes. By representing the soft hand's state through geometric mesh deformations rather than raw sensor values, KineSoft leverages the natural compliance of soft robots for dexterous skill learning while maintaining the precision necessary for complex manipulation tasks.

4.1 MOE Soft Robot End-effector

We leverage the multifinger omnidirectional end-effector (MOE) soft robot platform [1], which comprises modular finger units that each operate independently. Two servo motors actuate each finger by applying tension to four tendons (Figure 2A). MOE has a modular design that allows the fingers to be rearranged into various configurations to suit specific task requirements. In experiments, we use a three-finger variant, inspired by research on object controllability using three-fingered rigid end-effectors [49].

Building on the original MOE finger design [1], we propose embedding low-cost conductive elastic rubber directly into the silicone elastomer body of each finger. The conductive rubber acts as a sensor to measure deformation by varying its electrical resistance as it stretches, providing real-time proprioceptive feedback. Each finger incorporates four of these sensors, compactly positioned

between the tendons. A data acquisition (DAQ) circuit and board connected to the sensors can record resistance readings at approximately 400 Hz. By seamlessly integrating the sensors into the elastomer during fabrication, we developed a fully deformable and sensorized finger body. The combined state spaces from all twelve (12) strain sensors provides our estimate $\mathbf{S}$ of the soft robot hand's true deformation state space $\mathbf{M}$ (Section 3).

4.2 Shape Estimation Model

The shape estimation model maps strain sensor readings $\mathbf{R} \in \mathbb{R}^n$ to vertex displacements of the MOE fingers. Each of the three fingers contains four embedded strain sensors, giving $n = 12$ total resistance values. The model learns a function f that predicts per-vertex deformations conditioned on both the sensor input and the undeformed mesh:

$$f(\mathbf{R}, \{\mathbf{V}_{j,0}\}_{j=1}^{3}) = \{\Delta\mathbf{V}_j\}_{j=1}^{3},$$

where $\mathbf{V}_{j,0} \in \mathbb{R}^{N\times 3}$ denotes the initial vertex positions of finger j, and $\Delta\mathbf{V}_j$ are the predicted displacements. We implement f using a FoldingNet-based architecture [50, 21]. For each finger j, an encoder h_{enc} transforms its four resistance values $\mathbf{R}_j$ into a latent code $\mathbf{z}_j \in \mathbb{R}^{128}$ as $\mathbf{z}_j = h_{\text{enc}}(\mathbf{R}_j)$.

A decoder h_{dec} then predicts the displacement of each vertex $\mathbf{v}_{j,0}^i$ by combining its initial position with $\mathbf{z}_j$:

$$\Delta\mathbf{v}_{j,i} = h_{\text{dec}}(\mathbf{v}_{j,0}^i, \mathbf{z}_j), \quad \mathbf{v}_{j,t}^i = \mathbf{v}_{j,0}^i + \Delta\mathbf{v}_{j,i}.$$

This formulation allows the model to learn local deformation behavior conditioned on global strain input while preserving mesh topology. Predicting displacements rather than absolute positions improves stability and generalization as shown in comparison to baselines. The model is trained on a dataset of simulated deformations as described in the Appendix paired with corresponding simulated strain readings.

4.3 Sim-to-Real Domain Alignment

To bridge the gap between simulation and real-world deployment, we introduce a calibration procedure to align sensor readings from physical experiments with the simulated deformation model. Our goal is to determine a set of correction factors $\kappa_0, \ldots, \kappa_{n-1}$, one per sensor, that align the strain-based deformation estimates in simulation with real-world resistance measurements.

We derive the following objective, based on the sensor model described in the Appendix, to align simulated deformations with real-world sensor readings:

$$\underset{\kappa_0,\ldots,\kappa_{n-1}}{\operatorname{argmin}} \sum_{i=0}^{n-1}\sum_{t=0}^{T-1}\left(\sqrt{\frac{R_{i,t}}{R_{i,0}}} - \kappa_i \frac{L_{i,t}^S - L_{i,0}^S}{L_{i,0}^S} - 1\right)^2, \tag{1}$$

where n is the number of sensors, T is the number of sample frames, $R_{i,t}$ is the resistance measured from the i-th sensor at time step t, and $L_{i,t}^S$ is the corresponding simulated internal length of the i-th sensor at time t The correction factors κ_i are optimized to compensate for the domain mismatch between real and simulated sensor responses.

In practice, the true simulated lengths corresponding to each real-world resistance reading are unavailable, and the sensor-to-geometry mapping is not directly observable. Thus, we approximate this mapping by minimizing the unsupervised Chamfer distance between the observed 3D positions of points on the soft fingers and the predicted surface points from our shape estimation model:

$$\mathcal{L}_{\text{UCD}} = \sum_{j=0}^{m-1} \sum_{\mathbf{p}_{\text{obs}} \in \mathcal{P}_{\text{obs}}^{(j)}} \min_{\mathbf{p}_{\text{pred}} \in \mathcal{P}_{\text{pred}}^{(j)}} \|\mathbf{p}_{\text{obs}} - \mathbf{p}_{\text{pred}}\|^2, \tag{2}$$

where $\mathcal{P}_{\text{obs}}^{(j)}$ and $\mathcal{P}_{\text{pred}}^{(j)}$ denote the observed and predicted point clouds for the j-th example.

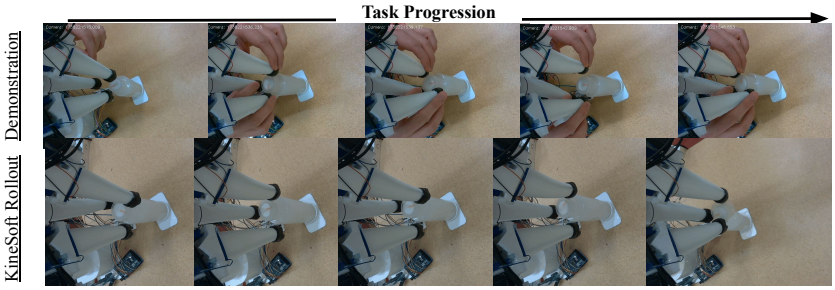

Figure 3: **Demonstration and KineSoft Rollout for Bottle Unscrewing Task.**

This alignment procedure ensures that the internal deformation estimated from resistance measurements produces a surface mesh consistent with external geometry observations. It connects to the overall pipeline by enabling the trained model to generalize from simulation to physical deployment without requiring paired real-world deformation labels.

4.4 Shape-conditioned Controller

The shape-conditioned controller leverages real-time proprioceptive mesh state estimation of the MOE fingers to execute desired shape trajectories. For each finger j, the controller compares the current estimated vertex positions $\mathbf{V}_t$ with desired target positions $\mathbf{V}_t^D$ generated from the policy trajectory. Each finger is actuated by a pair of antagonistic tendons controlled by two servos. The actuation directions for each servo pair are represented by unit vectors $\mathbf{d}_{2j}, \mathbf{d}_{2j+1} \in \mathbb{R}^2$ that capture the primary deformation modes. The servo adjustments $\delta u_{j,t}$ for each finger are computed by projecting the shape error onto these actuation directions:

$$\delta u_{j,t} = k_p \sum_n \mathbf{e}_{j,t}^n \cdot [\mathbf{d}_{2j}, \mathbf{d}_{2j+1}]^T,$$

where k_p is a scalar gain and $\mathbf{e}_{j,t}^n$ is the shape error for vertex n defined by $\mathbf{e}_{j,t} = \mathbf{V}_{t,j}^D - \mathbf{V}_{t,j}$. In deployment, the controller clips actions to prevent overloading the actuators. The controller runs at 100 Hz with the shape estimation at each step, enabling responsive shape trajectory tracking. By projecting shape errors onto fitted actuation directions, the controller effectively translates desired deformations into appropriate servo commands despite the complex relationship between tendon actuation and finger deformation.

4.5 Imitation Policy

The shape estimation model (Section 4.2) provides proprioceptive feedback via predicted surface vertex positions $\mathbf{V}_t$, while a wrist-mounted RGB-D camera captures external point cloud observations $\mathcal{P}_t$ of the workspace. We train a diffusion policy [11] to imitate manipulation skills using these complementary inputs.

The policy predicts actions $a_t = \{\Delta\mathbf{V}_t, \Delta\mathbf{p}_t\}$ that couple surface deformations and end-effector motion. The state $s_t = \{h_{\text{shape}}(\mathbf{V}_t), h_{\text{pc}}(\mathcal{P}_t), \mathbf{p}_t\}$ combines proprioceptive and exteroceptive information: h_{shape} is an MLP encoder applied to mesh vertices with temporal correspondence, h_{pc} is a DP3 encoder for the RGB-D point cloud, and $\mathbf{p}_t$ is the current end-effector pose.

The diffusion-based imitation policy learns to denoise a vertex-level action trajectory via the reverse process $a_{t-1} = \mu_\theta(a_t, s_t, t) + \sigma_t \mathbf{z}$, where μ_θ is a learned denoising network and $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$. Combined with the shape-conditioned controller (Section 4.4), the policy enables robust, contact-rich manipulation by fusing learned high-level intent with precise low-level tracking.

5 Experiments

We evaluate KineSoft through real-world experiments that assess each component of the framework and key design choices. We quantify the accuracy of the shape estimation model, followed

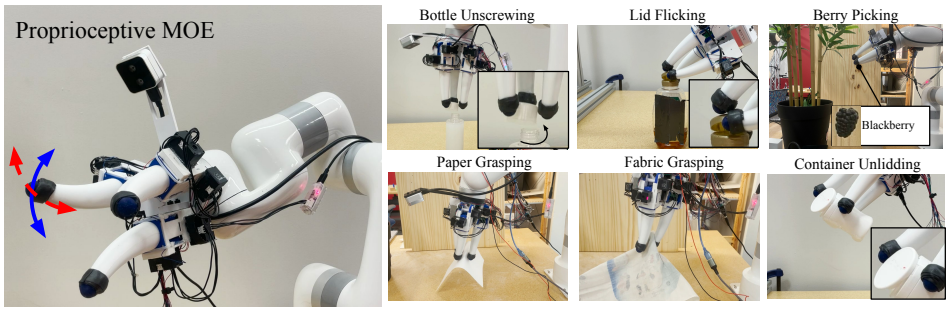

Figure 4: **Tasks.** We evaluate the performance of the shape-based KineSoft policy across six manipulation tasks. These tasks highlight the advantages of soft robotic hands in contact-rich and delicate object interaction. Red arrows indicate actuation directions for Servo 1; blue arrows indicate actuation directions for Servo 2.

by evaluating the shape-conditioned controller's ability to track target deformations. Finally, we assess KineSoft policies across six in-hand manipulation tasks involving both rigid and deformable objects. These experiments demonstrate the effectiveness of shape-based representations for proprioceptive control and reliable manipulation, particularly in settings where soft robots offer inherent advantages.

5.1 Shape Estimation

We evaluated the proposed shape estimation model against baselines from the literature: the constant curvature model [51, 43], an analytical representation of soft robot deformation, and DeepSoRo [52], a learning-based point cloud reconstruction method. In addition, we compared against two ablations of our approach: a naively calibrated linear mapping and an unconstrained fully connected model, both trained using the domain alignment procedure described in Section 4.3. Evaluation was conducted using observed point clouds captured from the physical setup shown in the Appendix, and performance was measured using the unidirectional Chamfer distance defined in Equation 2. Quantitative results are summarized in Table 1. Our model achieved a shape estimation error of 1.92,mm, representing a 41.3% improvement over the best baseline (DeepSoRo) and a 60.9% improvement over the linear variant of our method.

Table 1: Shape Estimation Fidelity

Method	Shape Error [mm]
PneuFlex Sensor	3.70 ± 1.36
DeepSoRo	3.27 ± 1.05
KineSoft (naive)	4.91 ± 2.85
KineSoft (unconstrained)	4.36 ± 3.47
KineSoft (model)	**1.92 ± 0.39**

5.2 Shape-conditioned Controller Performance

We evaluated the proposed shape-conditioned controller against a strain-tracking baseline that directly uses sensor readings for control as implemented in prior soft robot manipulation works [5, 53]. For evaluation, we collected reference trajectories through kinesthetic teaching, where a demonstrator physically deformed the fingers into desired configurations. During execution, the controllers had to track these trajectories using tendon actuation. This evaluation highlights a fundamental challenge in soft robot imitation: the sensor signals generated during kinesthetic demonstration (when fingers are manually deformed) differ significantly from those produced during autonomous execution (when tendons are actuated). The strain-tracking baseline, which attempts to directly match these sensor readings, struggles with this demonstration-execution gap, achieving 6.20 mm tracking error. In contrast, the proposed shape-conditioned controller bridges this gap by tracking the geometric shape itself rather than the underlying sensor signals, achieving 3.29 mm error, a 47 % improvement.

Table 2: Shape Tracking Error Comparison

Method	Representation	Error [mm]
Strain-tracking	Strain	6.20 ± 2.39
KineSoft	Mesh	**3.29 ± 0.91**

Table 3: Task success rates on 6 different tasks over 20 trials each comparing the baseline strain-matching method to the proposed KineSoft.

Method	Bottle Unscrewing	Lid Flicking	Berry Picking	Paper Grasping	Fabric Pickup	Container Unlidding
Strain Policy	0/20	18/20	7/20	13/20	1/20	3/20
KineSoft	**17/20**	**20/20**	**16/20**	**19/20**	**13/20**	**14/20**

5.3 In-hand Manipulation Task Performance

We evaluated KineSoft on six in-hand manipulation tasks (Figure 4) to showcase both the strengths of soft robots, such as delicate object handling [54, 55, 56] and contact-rich interaction [1, 57], and their conventional challenges, including tasks requiring accurate multi-finger coordination [58].

Bottle Unscrewing: MOE must rotate and lift a bottle cap using coordinated finger motion.

Lid Flicking: MOE flicks open a hinged lid on a container.

Berry Picking: MOE gently picks a berry from a soft branch, evaluating precise grasping and compliant contact with fragile objects inspired by prior works on soft robot fruit harvesting [56].

Paper Grasping: MOE initiates contact with and lifts a flat sheet of paper from a table.

Fabric Grasping: MOE conforms around and lifts a loosely draped piece of fabric, highlighting contact-rich manipulation of deformable objects.

Container Unlidding: MOE grasps and opens the snap-top lid of a plastic container while maintaining a stable grasp. The task highlights multi-finger coordination and localized force application.

Each task was performed in 20 real-world trials using demonstrations collected via kinesthetic teaching, where the operator directly deforms the soft fingers. During execution, KineSoft reproduces these deformations through tendon actuation, bridging the sim-to-real gap with learned shape-aware control. Table 3 reports success rates across all six tasks. KineSoft achieved an 85 % success rate on Bottle Unscrewing, compared to 0 % for a strain-matching baseline based on a state-of-the-art soft in-hand manipulation method [53], implemented with a diffusion policy and trained on the same demonstration data. On Lid Flicking, KineSoft reached 100 % success, while the baseline achieved 90 %. For contact-sensitive tasks such as Berry Picking, Fabric Pickup, and Container Unlidding, KineSoft achieved success rates between 70 % and 80 %, whereas the baseline remained below 35 % on all three. These results highlight the effectiveness of shape-based representations for robust control and reinforce the advantages of soft hands in contact-rich and delicate manipulation settings.

6 Conclusion and Lessons

This paper presents KineSoft, a framework for learning dexterous manipulation skills with soft robot hands that embraces, rather than fights against, their inherent compliance. The key insight is recognizing that while this compliance enables intuitive kinesthetic teaching, it creates a fundamental gap between demonstration and execution, where the deformations and sensor signals during human demonstration differ substantially from those during autonomous execution through tendon actuation.KineSoft addresses this challenge through a hierarchical approach. A shape estimation model provides consistent geometric representations across demonstration and execution modes while a domain alignment method enables robust transfer of simulation-trained models to real hardware and a shape-conditioned controller reliably tracks the policy's generated deformation trajectories despite the different underlying actuation mechanisms. Then, a high-level imitation policy learns to generate target vertex deformations from demonstrations, capturing the intended manipulation strategy in geometric grounding. In experiments, results demonstrate that this shape-based hierarchical approach enables more effective skill transfer than methods that attempt to directly match sensor signals or joint configurations. This work suggests that successful imitation learning for dexterous soft robots requires careful consideration of how demonstration and execution modes differ, and appropriate intermediate representations to bridge this gap.

7 Limitations

While KineSoft demonstrates effective shape estimation and trajectory tracking for soft robotic manipulation, several limitations remain. First, the current system lacks explicit force-feedback mechanisms, relying instead on kinematic trajectories and deformation states to encode manipulation skills. This limits precise force control during object interactions. In contrast, human manipulation integrates both proprioception and tactile sensing, suggesting potential benefits from adding complementary tactile input.

Additionally, although kinesthetic teaching provides a robust approach to demonstrating some dexterous tasks, it requires the expert demonstrator to recognize the MOE finger's workspace limitations to avoid demonstrating shapes that the MOE cannot reach. In future work, we aim to leverage KineSoft's real-time shape estimation to communicate robot affordances to users via a visual user interface during demonstrations.

Although the sensors used in this work are readily available and low-cost, they introduce fabrication complexity when integrating into the prior MOE design, which was relatively simple to fabricate. We also observed that the sensors may not respond dynamically in fast manipulation tasks due to commonly observed soft body phenomena such as creep and hysteresis, where the sensor signals require some time to reach equilibrium conditions when stretched quickly. For durability, we included results on long-term continuous operation and sensor signal deviation in the Appendix.

KineSoft's shape-conditioned low-level controller weighs the errors from the vertices of the reference meshes equally. Implicitly, the controller prioritizes the vertices closer to the tip higher because the displacement tends to be greater further away from the base. For many tasks, this is an appropriate weighing as the finger tips play a vital role in performing dexterous in-hand manipulation tasks [59]. However, some tasks may require different regions of the fingers to be prioritized based on contact probability.

References

[1] U. Yoo, N. Dennler, E. Xing, M. Matarić, S. Nikolaidis, J. Ichnowski, and J. Oh. Soft and compliant contact-rich hair manipulation and care. In *Proc. of IEEE International Conference on Human-Robot Interaction (HRI)*, 2025.

[2] K. Gilday, I. Zubak, A. Raabe, and J. Hughes. From rigid to soft robotic approaches for minimally invasive neurosurgery. *arXiv preprint arXiv:2404.14071*, 2024.

[3] U. Yoo, N. Dennler, M. Mataric, S. Nikolaidis, J. Oh, and J. 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.

[4] C. d. Pasquier, J. Grannen, C. Pan, S. L. Huber, A. Smith, M. Kennedy, S. Song, D. Sadigh, and A. M. Okamura. A study of perceived safety for soft robotics in caregiving tasks. *arXiv preprint arXiv:2503.20916*, 2025.

[5] A. Bhatt, A. Sieler, S. Puhlmann, and O. Brock. Surprisingly robust in-hand manipulation: An empirical study. *arXiv preprint arXiv:2201.11503*, 2022.

[6] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus. Robust proprioceptive grasping with a soft robot hand. *Autonomous robots*, 43:681–696, 2019.

[7] Y. Yao, U. Yoo, J. Oh, C. G. Atkeson, and J. Ichnowski. Soft robotic dynamic in-hand pen spinning. *arXiv preprint arXiv:2411.12734*, 2024.

[8] C. Firth, K. Dunn, M. H. Haeusler, and Y. Sun. Anthropomorphic soft robotic end-effector for use with collaborative robots in the construction industry. *Automation in Construction*, 138: 104218, 2022.

[9] S. Abondance, C. B. Teeple, and R. J. Wood. A dexterous soft robotic hand for delicate in-hand manipulation. *IEEE Robotics and Automation Letters*, 5(4):5502–5509, 2020.

[10] R. Jitosho, C. E. Winston, S. Yang, J. Li, M. Ahlquist, N. J. Woehrle, C. K. Liu, and A. M. Okamura. Flying vines: Design, modeling, and control of a soft aerial robotic arm. *arXiv preprint arXiv:2503.20754*, 2025.

[11] 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*, page 02783649241273668, 2023.

[12] M. Memmel, J. Berg, B. Chen, A. Gupta, and J. Francis. STRAP: Robot sub-trajectory retrieval for augmented policy learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/pdf?id=4VHiptx7xe.

[13] Y. Hu, Q. Xie, V. Jain, J. Francis, J. Patrikar, N. Keetha, S. Kim, Y. Xie, T. Zhang, H.-S. Fang, et al. Toward general-purpose robots via foundation models: A survey and meta-analysis. *arXiv preprint arXiv:2312.08782*, 2023.

[14] 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.

[15] U. Yoo, A. Hung, J. Francis, J. Oh, and J. Ichnowski. Ropotter: Toward robotic pottery and deformable object manipulation with structural priors. In *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pages 843–850. IEEE, 2024.

[16] H. Qi, B. Yi, M. Lambeta, Y. Ma, R. Calandra, and J. Malik. From simple to complex skills: The case of in-hand object reorientation. *arXiv preprint arXiv:2501.05439*, 2025.

[17] J. Wang, Y. Yuan, H. Che, H. Qi, Y. Ma, J. Malik, and X. Wang. Lessons from learning to spin "pens". In *CoRL*, 2024.

[18] Y. Qin, W. Yang, B. Huang, K. Van Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. *arXiv preprint arXiv:2307.04577*, 2023.

[19] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy. *arXiv preprint arXiv:2403.03954*, 2024.

[20] L. Wang, J. Lam, X. Chen, J. Li, R. Zhang, Y. Su, and Z. Wang. Soft robot proprioception using unified soft body encoding and recurrent neural network. *Soft Robotics*, 10(4):825–837, 2023.

[21] U. Yoo, H. Zhao, A. Altamirano, W. Yuan, and C. Feng. Toward zero-shot sim-to-real transfer learning for pneumatic soft robot 3d proprioceptive sensing. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 544–551. IEEE, 2023.

[22] U. Yoo, Z. Lopez, J. Ichnowski, and J. Oh. Poe: Acoustic soft robotic proprioception for omnidirectional end-effectors. *arXiv preprint arXiv:2401.09382*, 2024.

[23] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

[24] C. Della Santina, C. Duriez, and D. Rus. Model-based control of soft robots: A survey of the state of the art and open challenges. *IEEE Control Systems Magazine*, 43(3):30–65, 2023.

[25] T. G. Thuruthel, E. Falotico, F. Renda, and C. Laschi. Model-based reinforcement learning for closed-loop dynamic control of soft robotic manipulators. *IEEE Transactions on Robotics*, 35 (1):124–134, 2018.

[26] P. Schegg, E. Ménager, E. Khairallah, D. Marchal, J. Dequidt, P. Preux, and C. Duriez. Sofagym: An open platform for reinforcement learning based on soft robot simulations. *Soft Robotics*, 10(2):410–430, 2023.

[27] S. Bhagat, H. Banerjee, Z. T. Ho Tse, and H. Ren. Deep reinforcement learning for soft, flexible robots: Brief review with impending challenges. *Robotics*, 8(1):4, 2019.

[28] E. Johns. Coarse-to-fine imitation learning: Robot manipulation from a single demonstration. In *IEEE international conference on robotics and automation (ICRA)*, pages 4613–4619. IEEE, 2021.

[29] S. P. Arunachalam, I. Güzey, S. Chintala, and L. Pinto. Holo-dex: Teaching dexterity with immersive mixed reality. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5962–5969. IEEE, 2023.

[30] S. P. Arunachalam, S. Silwal, B. Evans, and L. Pinto. Dexterous imitation made easy: A learning-based framework for efficient dexterous manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5954–5961. IEEE, 2023.

[31] D. Wei and H. Xu. A wearable robotic hand for hand-over-hand imitation learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 18113–18119. IEEE, 2024.

[32] A. I. Weinberg, A. Shirizly, O. Azulay, and A. Sintov. Survey of learning-based approaches for robotic in-hand manipulation. *Frontiers in Robotics and AI*, 11:1455431, 2024.

[33] D. Rus and M. T. Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553): 467–475, 2015.

[34] X. Zhou, X. Chen, and T. Yang. Soft robotic grippers. *Advanced Intelligent Systems*, 5(1): 2000198, 2023.

[35] F. Liu, K. Puthuveetil, A. Padmanabha, K. Khokar, Z. Temel, and Z. Erickson. Skingrip: An adaptive soft robotic manipulator with capacitive sensing for whole-limb bathing assistance. *arXiv preprint arXiv:2405.02772*, 2024.

[36] S. Puhlmann, J. Harris, and O. Brock. Rbo hand 3: A platform for soft dexterous manipulation. *IEEE Transactions on Robotics*, 38(6):3434–3449, 2022.

[37] O. Yasa, Y. Toshimitsu, M. Y. Michelis, L. S. Jones, M. Filippi, T. Buchner, and R. K. Katzschmann. An overview of soft robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 6(1):1–29, 2023.

[38] A. Gupta, C. Eppner, S. Levine, and P. 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.

[39] A. Pagoli, F. Chapelle, J. A. Corrales, Y. Mezouar, and Y. 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.

[40] S. Zhou, Y. Li, Q. Wang, and Z. Lyu. Integrated actuation and sensing: Toward intelligent soft robots. *Cyborg and Bionic Systems*, 5:0105, 2024.

[41] S. Ku, B.-H. Song, T. Park, Y. Lee, and Y.-L. Park. Soft modularized robotic arm for safe human–robot interaction based on visual and proprioceptive feedback. *The International Journal of Robotics Research*, 43(8):1128–1150, 2024.

[42] F. Stella, Q. Guan, C. Della Santina, and J. Hughes. Piecewise affine curvature model: a reduced-order model for soft robot-environment interaction beyond pcc. In *IEEE International Conference on Soft Robotics (RoboSoft)*, pages 1–7. IEEE, 2023.

[43] U. Yoo, Y. Liu, A. D. Deshpande, and F. Alamabeigi. Analytical design of a pneumatic elastomer robot with deterministically adjusted stiffness. *IEEE robotics and automation letters*, 6 (4):7773–7780, 2021.

[44] V. Wall, G. Zöller, and O. Brock. A method for sensorizing soft actuators and its application to the rbo hand 2. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4965–4970. IEEE, 2017.

[45] Y. Liu, U. Yoo, S. Ha, S. F. Atashzar, and F. Alambeigi. Influence of antagonistic tensions on distributed friction forces of multisegment tendon-driven continuum manipulators with irregular geometry. *IEEE/ASME Transactions on Mechatronics*, 27(5):2418–2428, 2021.

[46] Y. Liu, S. H. Ahn, U. Yoo, A. R. Cohen, and F. Alambeigi. Toward analytical modeling and evaluation of curvature-dependent distributed friction force in tendon-driven continuum manipulators. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8823–8828. IEEE, 2020.

[47] J. Tapia, E. Knoop, M. Mutnỳ, M. A. Otaduy, and M. Bächer. Makesense: Automated sensor design for proprioceptive soft robots. *Soft robotics*, 7(3):332–345, 2020.

[48] R. K. Katzschmann, C. Della Santina, Y. Toshimitsu, A. Bicchi, and D. Rus. Dynamic motion control of multi-segment soft robots using piecewise constant curvature matched with an augmented rigid body model. In *IEEE International Conference on Soft Robotics (RoboSoft)*, pages 454–461. IEEE, 2019.

[49] M. T. Mason and J. K. Salisbury Jr. Robot hands and the mechanics of manipulation. *The MIT Press, Cambridge, MA*, 1985.

[50] Y. Yang, C. Feng, Y. Shen, and D. Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 206–215, 2018.

[51] C. Della Santina, A. Bicchi, and D. Rus. On an improved state parametrization for soft robots with piecewise constant curvature and its use in model based control. *IEEE Robotics and Automation Letters*, 5(2):1001–1008, 2020.

[52] R. Wang, S. Wang, S. Du, E. Xiao, W. Yuan, and C. Feng. Real-time soft body 3d proprioception via deep vision-based sensing. *IEEE Robotics and Automation Letters*, 5(2):3382–3389, 2020.

[53] A. Sieler and O. 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.

[54] N. R. Sinatra, C. B. Teeple, D. M. Vogt, K. K. Parker, D. F. Gruber, and R. J. Wood. Ultragentle manipulation of delicate structures using a soft robotic gripper. *Science Robotics*, 4(33): eaax5425, Aug. 2019. doi:10.1126/scirobotics.aax5425. URL `https://doi.org/10.1126/scirobotics.aax5425`. Publisher: American Association for the Advancement of Science.

[55] D. Bauer, C. Bauer, and N. S. Pollard. Soft robotic end-effectors in the wild: A case study of a soft manipulator for green bell pepper harvesting. In *2nd AAAI Workshop on AI for Agriculture and Food Systems*, 2023.

[56] A. L. Gunderman, J. A. Collins, A. L. Myers, R. T. Threlfall, and Y. Chen. Tendon-Driven Soft Robotic Gripper for Blackberry Harvesting. *IEEE Robotics and Automation Letters*, 7(2): 2652–2659, Apr. 2022. ISSN 2377-3766. doi:10.1109/LRA.2022.3143891.

[57] E. Páll and O. Brock. Analysis of open-loop grasping from piles. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2591–2597. IEEE, 2021.

[58] J. Hughes, U. Culha, F. Giardina, F. Guenther, A. Rosendo, and F. Iida. Soft manipulators and grippers: A review. *Frontiers in Robotics and AI*, 3:69, 2016.

[59] I. M. Bullock, T. Feix, and A. M. Dollar. Analyzing human fingertip usage in dexterous precision manipulation: Implications for robotic finger design. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1622–1628. IEEE, 2014.

[60] O. Starkova and A. Aniskevich. Poisson's ratio and the incompressibility relation for various strain measures with the example of a silica-filled sbr rubber in uniaxial tension tests. *Polymer Testing*, 29(3):310–318, 2010.

[61] A. Zhang, T.-H. Wang, R. L. Truby, L. Chin, and D. Rus. Machine learning best practices for soft robot proprioception. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2564–2571. IEEE, 2023.

[62] J. Westwood et al. Sofa–an open source framework for medical simulation. *Medicine Meets Virtual Reality 15: In Vivo, in Vitro, in Silico: Designing the Next in Medicine*, 125:13, 2007.

A Appendix

A.1 Task Frames

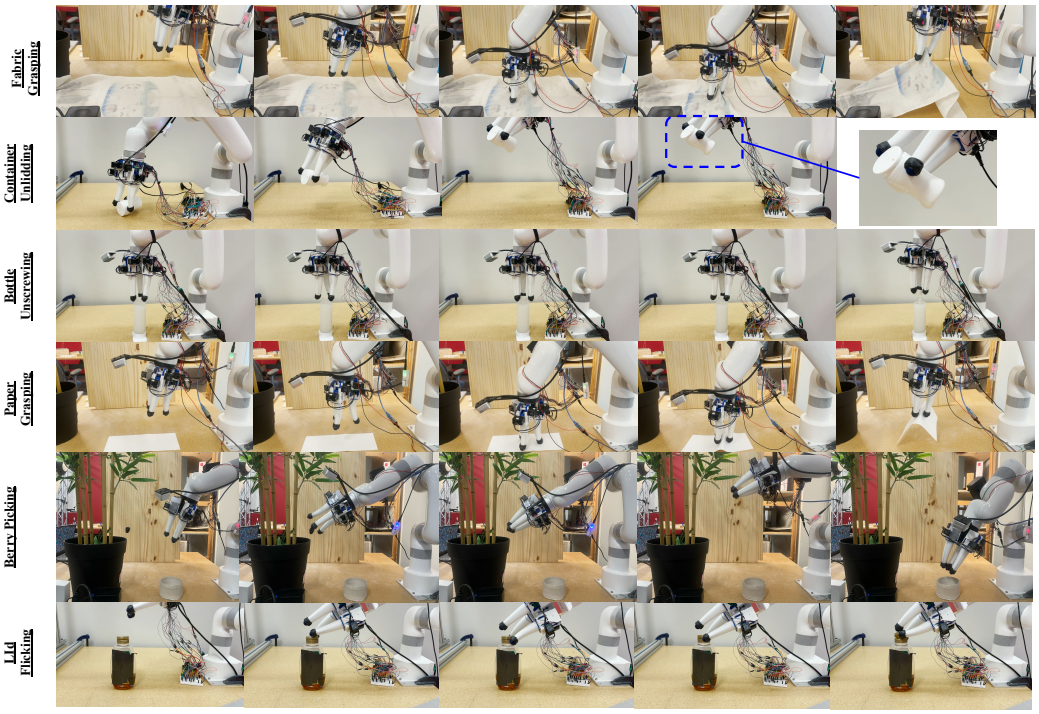

Figure 5: **Tasks**

A.2 Shape Estimation and Tracking Evaluation

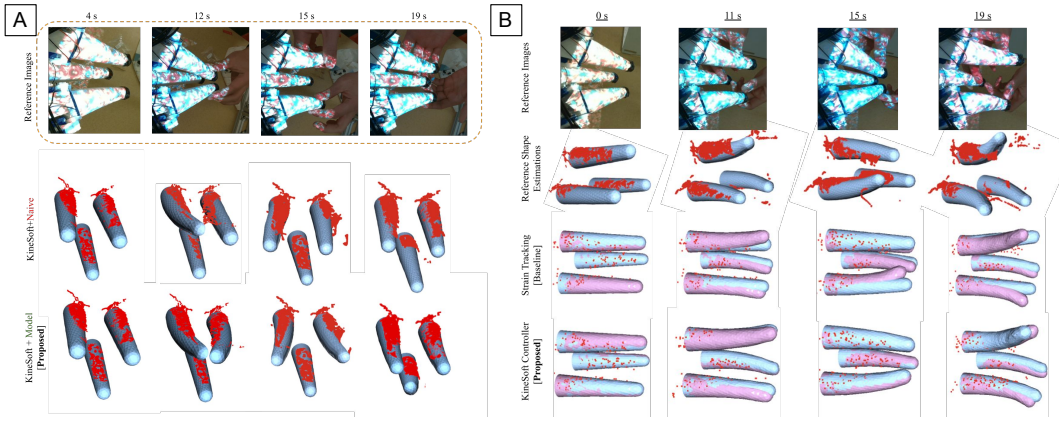

Figure 6: **Shape estimation and trajectory tracking performance evaluation.** We provide each of the shape estimation models and controllers with kinesthetically deformed shape trajectories. A: Shape estimation model comparisons with real-world ground-truth data (red points). B: Shape tracking comparisons with the real-world ground-truth data (red points), references shapes (red), and achieved shapes (blue).

A.3 Details on Sensor Model

We assume the embedded sensors are perfectly incompressible and isotropic, a common assumption in soft body mechanics for highly elastic rubber, particularly when infused with particle fillers [60]. These fillers, like those used in the off-the-shelf conductive rubbers embedded in MOE, enable the sensors to exhibit changes in resistivity when stretched. The sensors have a cylindrical shape, so we model the relationship between the cross-sectional area and the strain in the axial direction for sensor i at time $t \geq 0$ with the incompressibility assumption as:

$$L_{i,0}A_{i,0} = L_{i,t}A_{i,t}, \tag{3}$$

where $L_{i,0}$ and $A_{i,0}$ are the initial length and cross-sectional area; $L_{i,t}$ and $A_{i,t}$ are the corresponding values at time t.

For conductive materials, resistance generally has a linear relationship with strain. The observed resistance for the sensor indexed at i is given by:

$$R_{i,t} = \rho_i \frac{L_{i,t}}{A_{i,t}}, \tag{4}$$

where ρ_i is the conductivity factor, assumed to be constant for sensor i across time. Relating Equation 3 and Equation 4, we derive:

$$\sqrt{\frac{R_{i,t}}{R_{i,0}}} - 1 = \frac{L_{i,t} - L_{i,0}}{L_{i,0}}. \tag{5}$$

This relationship is independent of the material conductivity ρ_i, enabling a direct mapping from observed resistance to strain. However, in real-world applications, fabrication imperfections, such as connecting wires to the DAQ boards, can introduce errors into the initial length of the embedded sensors. These imperfections result in a deviation between the real sensor lengths ($L^R_{i,0}$, $L^R_{i,t}$) and simulated sensor lengths ($L^S_{i,0}$, $L^S_{i,t}$):

$$L^R_{i,0} = L^S_{i,0} + \epsilon_i, \quad L^R_{i,t} = L^S_{i,t} + \epsilon_i,$$

where ϵ_i is a constant error specific to each sensor i. This error propagates to the strain relationship as:

$$\frac{L^R_{i,t} - L^R_{i,0}}{L^R_{i,0}} = \frac{1}{1 + \frac{\epsilon_i}{L^S_{i,0}}} \cdot \frac{L^S_{i,t} - L^S_{i,0}}{L^S_{i,0}}. \tag{6}$$

The constant factor $\frac{1}{1+\frac{\epsilon_i}{L^S_{i,0}}}$ can be denoted as $\kappa_i \in \kappa$, representing the constant correction factor for sensor i. Substituting this into Equation 5, we obtain:

$$\sqrt{\frac{R_{i,t}}{R_{i,0}}} - 1 = \kappa_i \frac{L^S_{i,t} - L^S_{i,0}}{L^S_{i,0}}, \tag{7}$$

where the observed resistances $R_{i,t}, R_{i,0}$ are measured with the DAQ setup. For the n embedded sensors, aligning the simulated and observed distributions involves optimizing the constant correction parameters $\kappa_0, \kappa_1, \ldots, \kappa_{n-1}$.

A.4 Baselines

For shape estimation, we compare with analytical and learning-based baselines:

Constant curvature model [51, 43]. Constant curvature model is a common representation for the continuum deformation behavior of a soft robot that parametrizes the shape with a single curvature curve [61]. Typically, the independent parameters of the state of the robot are defined by r_{curve} and θ_{curve}. Assuming a constant length, L_{curve} of the robot, we get the constraint:

$$L_{\text{curve}} = r_{\text{curve}}\theta_{\text{curve}}.$$

Algorithm 1 Domain Alignment Optimization

Input: Shape predictor f_θ, transformations $\{T\}$, observations $\mathcal{P}_{\text{obs}}$, initial resistances R_0
  Initialize $\theta = \{\kappa \in \mathbb{R}^{24}, \phi \in \mathbb{R}^3\} \leftarrow \mathbf{0}$
  Initialize CMA-ES optimizer $\mathcal{O}(\theta)$
  while not converged **do**
    Sample candidates $\{\theta\} \sim \mathcal{O}$
    for each θ **do**
      $\Delta R \leftarrow \sqrt{R/R_0} - 1$ ▷ Strain model for resistances
      Apply correction: $S \leftarrow \Delta R \cdot \kappa_{\mathbb{I}[\Delta R<0]}$
      $\mathbf{V} \leftarrow f_\theta(S)$ ▷ Predict vertices
      $\mathcal{P}_{\text{pred}} \leftarrow \bigcup \mathbf{V} \cdot T \cdot [R_y(\phi)|\mathbf{0}]$ ▷ Combine
      $\ell \leftarrow \mathcal{L}_{\text{UCD}}(\mathcal{P}_{\text{obs}}, \mathcal{P}_{\text{pred}})$ ▷ Compute error
    end for
    Update $\mathcal{O}$ with candidates and losses
  end while
  return θ^* with minimum loss

In typical applications, additional term ϕ_{curve} is introduced to represent the plane of bending [48]. We implemented this simplified representation for soft robot shape using the proposed strain model as outlined in Section 4.3 and fitting r_{curve} and θ_{curve} to the observed strains in each side of the curve. We transformed the cross-section boundary to the curve during the evaluation and measured the chamfer distance to the reference.

DeepSoRo [52]. DeepSoRo architecture deploys a FoldingNet [50] decoder conditioned on visual observations to predict the current shape of a deformable body. Crucially, it is trained with chamfer distance and originally trained on partial real-world shape observations, resulting in partial point cloud reconstruction outputs without frame-to-frame correspondences. Additionally, the model directly outputs the point cloud positions in contrast to KineSoft, which learns a deformation field and produces vertex displacement with frame-to-frame correspondences. We augment DeepSoRo for evaluation by training the model on KineSoft's simulated training data and using the proposed domain alignment process.

Shape-tracking Baselines. For shape tracking and task performance evaluations we provide the results against **Strain Policy**. Strain policy, based on prior works that directly use sensor readings without intermediate representations for learning manipulation policies [53], uses raw sensor measurements instead of reconstructed shapes. For shape tracking evaluation, we modified the low-level controller from Section 4.4 to track reference sensor readings directly through proportional tendon actuation. For task performance evaluation, we trained a diffusion policy using the same 50 demonstrations we use for KineSoft, but with raw strain signals and wrist-mounted camera observations as input states.

A.5 Data Generation and Shape Estimation Model Training

To train the model, we generate a large dataset of deformed meshes using SOFA (Simulation Open Framework Architecture) [62]. We simulate a tetrahedral finite-element mesh of the MOE finger with a Neo-Hookean hyperelastic material model parameterized by elastic material properties that are randomized at runtime.

We model the tendon actuation with massless, inextensible cables running through a series of fixed points within the finger body. We discretize each tendon path to segments defined by 3D attachment points embedded in the tetrahedral mesh. The cable constraint applies forces to these points to maintain constant length while allowing sliding, effectively simulating the mechanical behavior of Bowden cable transmission. The soft body scene is solved with an implicit Euler time integration scheme and uses a conjugate gradient solver for the system matrices. We generate training data by randomly sampling tendon actuation commands within the feasible range and recording the resulting deformed vertex positions and embedded sensor strains. To simulate rich deformation behaviors

including contact-like effects, we apply random external forces to the finger surface. These forces are randomly applied over time with sufficiently large radii to ensure smooth deformations that mimic natural contact interactions, without requiring explicit and difficult-to-model contacts in the scene.

We train the model using a mean squared error (MSE) loss on vertex displacements:

$$\mathcal{L} = \frac{1}{3N} \sum_{j=1}^{3} \sum_{i=1}^{N} \|\Delta \mathbf{v}_{j,i} - \Delta \mathbf{v}^*_{j,i}\|^2,$$

where $\Delta \mathbf{v}^*_{j,i}$ represents the ground-truth displacement for vertex i of mesh finger j. This choice of loss function provides strong supervision by enforcing explicit vertex-wise correspondence between predicted and ground-truth meshes. Because we leverage simulated data to train the model, we can exploit the vertex-level correspondences in the meshes. We contrast this to prior works that relied on chamfer distance loss over real-world partial observations [52], MSE loss ensures that each vertex learns to track its specific local deformation patterns, enabling precise reconstruction of the full finger shape.

A.6 Experiment Setup

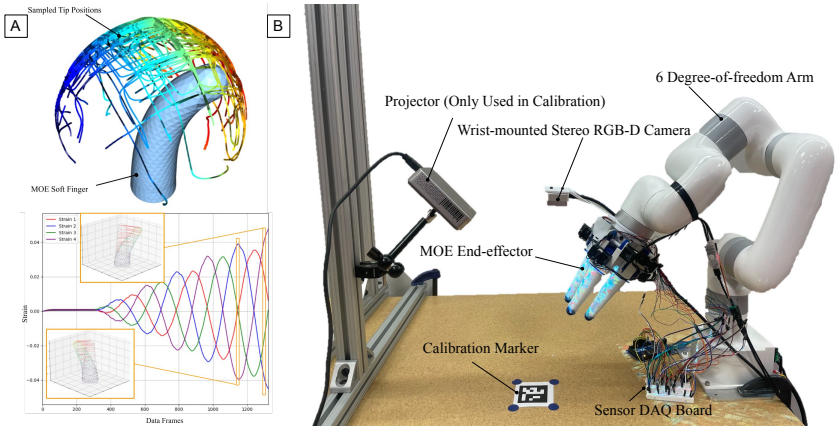

Figure 7: **Simulation and real-world setup**. A: Simulated robot workspace and sample of simulated strain signals. B: Real-world robot setup with the projected patterns to improve ground-truth shape observations for evaluation and calibration.

A.7 Failiure Cases

We conducted stress testing through 1,000 continuous servo cycles over 14 hours, observing maximum signal drift of 8.3% in a sensor (Fig. 8) that resulted in 0.062 ± 0.0056 mm mean vertex error. After rest, the signals returned to baseline levels and our original manipulation experiments were conducted over several weeks without recalibration. We appreciate the reviewers highlighting this important concern and will update the manuscript with these discussions on long-term sensor reliability.

KineSoft failures occurred mainly when fingers lost or made unintended contact during manipulation transitions—a limitation of kinematic imitation without tactile sensing, which we will highlight in the limitations section. Baseline strain policies failed significantly more, as demonstration strain patterns often cannot be directly reproduced through tendon actuation. The results support KineSoft's approach of using shape as an intermediate representation between demonstration and execution modes.

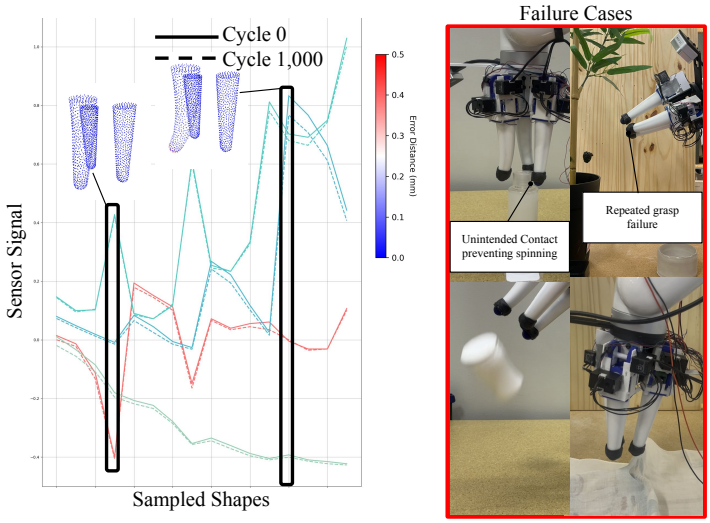

Figure 8: **Left:** Sensor signal degradation after 1000 cycles. **Right:** Failiure cases for manipulation tasks.

A.8 Sensor Signals

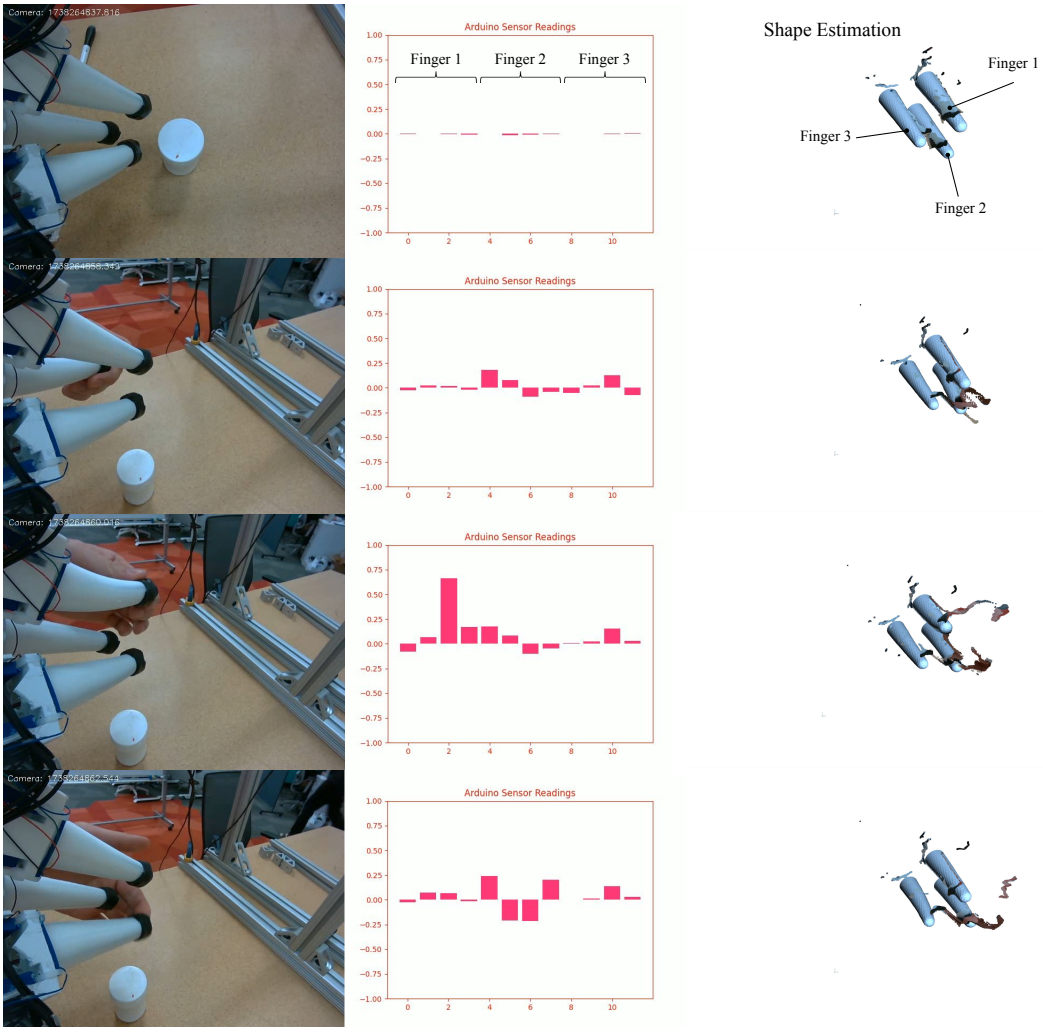

Figure 9: **Sensor signals and corresponding shape estimation**