2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Real-Time Trajectory Optimization for Continuum Robots in Human–Robot Interaction Using Vision-Based Target Pose Estimation

***Abstract*—Continuum robots possess intrinsic compliance, high flexibility, and continuously deformable structures, making them well-suited for safe human–robot interaction (HRI). However, their continuous backbone and high degrees of freedom pose significant challenges for real-time trajectory generation: motions must satisfy curvature constraints while adapting to uncertain and rapidly changing human inputs. Existing methods can generate smooth and feasible paths, but many are computationally intensive, neglect curvature continuity or mechanical constraints, or lack adaptability to dynamic environments. As a result, producing smooth, feasible, and responsive trajectories for continuum robots in interactive scenarios remains challenging. To address this, we propose a real-time trajectory optimization framework that integrates temporally filtered, vision-based human intention signals with curvature-constrained planning. Human hand motions are converted into stable reference signals, which guide a sliding-window sequential quadratic programming (SQP) optimizer. The planner continuously generates smooth and feasible trajectories that adapt in real time to evolving inputs. Simulations and hardware experiments demonstrate accurate tracking, robustness to noise, and timely adaptation, highlighting the framework's potential to enable safe and natural human–continuum robot collaboration in real-world applications.**

***Index Terms*—Continuum robots, trajectory optimization, human-robot interaction, vision-based intention estimation, real-time motion planning**

I. Introduction

Continuum robots have attracted increasing interest due to their intrinsic compliance, high dexterity, and smooth bending capability, which together make them ideal for human–robot interaction (HRI) scenarios [1]–[4]. Unlike rigid-link manipulators, continuum robots deform continuously along their backbone, enabling safe contact with humans and dexterous operation in cluttered or confined spaces. These characteristics provide unique advantages for collaborative applications such as surgical assistance, teleoperation, and shared workspaces, where safety and adaptability are paramount.

Despite these advantages, enabling continuum robots to interact naturally with humans remains highly challenging. First, their continuous morphology leads to infinite-dimensional configuration spaces and nonlinear kinematics, making real-time trajectory optimization substantially more complex than for rigid manipulators [5]–[8]. Second, mechanical constraints such as bounded curvature must be enforced to avoid physically infeasible or unsafe motions. Third, when operating in HRI scenarios, the robot must continuously adapt to human motion goals that are uncertain, noisy, and rapidly changing; however, most existing approaches fail to balance the requirements of constraint-aware feasibility and real-time responsiveness to human inputs [9], [10].

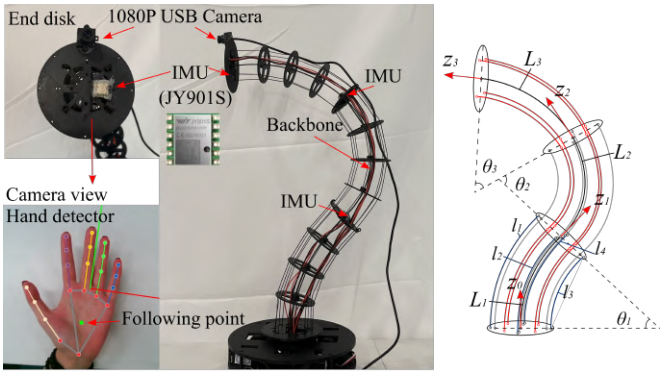

Fig. 1. (a) Continuum robot prototype. (b) Kinematic model with cable lengths.

To address these challenges, many collaborative tasks require capturing the immediate, task-relevant objectives of human motion. In this work, we focus on estimating the short-term target pose of the human collaborator from vision. These target pose signals serve as a direct and practical input for guiding continuum robot trajectories, enabling the robot to safely and responsively adapt to human motion in close-proximity human–robot interaction scenarios. Recent progress has been made in both human motion understanding and robot trajectory optimization, but their integration for continuum robots remains limited. A particularly intuitive and non-invasive way to infer human intention in close-proximity tasks is through vision-based pose estimation of the hand or body [11], [12]. Vision-based approaches can reliably capture hand and body motions in real time, providing a promising source of guidance for collaborative tasks [13]–[16]. However, leveraging these pose-level intention signals within continuum robot planning is nontrivial, since the robot's continuous backbone must satisfy curvature and feasibility constraints. At the same time, optimization-based planners have shown success in generating smooth and dynamically feasible trajectories for rigid manipulators [17]–[19], yet applying them to continuum robots often leads to

excessive computational cost or neglects curvature continuity [20]–[22]. This gap highlights the need for a unified framework that can translate noisy pose-level intention signals into real-time, constraint-aware trajectories for continuum robots [23].

In this paper, we propose a novel real-time trajectory optimization framework that directly incorporates vision-based pose-level intention estimation into curvature-constrained motion planning for continuum robots. Our approach formulates filtered target pose signals as references for a sliding-window sequential quadratic programming (SQP) optimizer, which continuously enforces curvature feasibility while adapting to evolving human inputs. This enables the robot to generate smooth, safe, and responsive motions suitable for close-proximity collaboration. The main contributions are:

- A tightly-integrated real-time framework that robustly integrates noisy, vision-based target pose estimates with sliding-window SQP trajectory optimization, ensuring continuous adaptation to dynamic human motion.
- Explicit enforcement of curvature and terminal pose constraints, generating physically feasible and smooth trajectories for continuum robots in close-proximity HRI scenarios.
- Extensive simulations and physical experiments under diverse conditions demonstrating the framework's effectiveness in producing safe, responsive, and adaptable motions under diverse human motion patterns.

This work bridges the gap between pose-level intention estimation and continuum robot motion planning, advancing the state of the art in human-centered continuum robot control.

II. Continuum Robot Modeling

We consider a single segment of a tendon-driven continuum robot actuated by four tendons placed at the corners of a square cross-section. Let L denote the arc length of the neutral backbone, and let the four tendon lengths be l_1, l_2, l_3, and l_4, as illustrated in Fig. 1. The angular positions of the tendons are defined as

$$\theta_j = \frac{\pi}{4} + (j-1)\frac{\pi}{2}, \qquad j = 1, 2, 3, 4, \tag{1}$$

which correspond to $\theta_1 = \pi/4$, $\theta_2 = 3\pi/4$, $\theta_3 = 5\pi/4$, $\theta_4 = 7\pi/4$. Each tendon is located at a radial distance r_t from the neutral axis, such that the half-side length of the square cross-section is given by $d = r_t/\sqrt{2}$.

A. Tendon lengths to curvature components

Define the tendon length deviations relative to the neutral backbone as $\Delta l_j = l_j - L$, $j = 1, \ldots, 4$, and denote the mean change as $\overline{\Delta l} = \frac{1}{4}\sum_{j=1}^{4} \Delta l_j$.

represents an axial (pure length) component; if the backbone is assumed inextensible we remove this component and use

$$\Delta l'_j = \Delta l_j - \overline{\Delta l}, \qquad j = 1, \ldots, 4. \tag{2}$$

Under the Piecewise Constant Curvature (PCC) assumption the local curvature vector can be represented by its Cartesian components in the local xy-plane

$$\boldsymbol{\kappa} = \begin{bmatrix} \kappa_x \\ \kappa_y \end{bmatrix} = \kappa \begin{bmatrix} \cos\beta \\ \sin\beta \end{bmatrix}, \tag{3}$$

where $\kappa = \sqrt{\kappa_x^2 + \kappa_y^2}$ is the curvature magnitude and β is the bending-plane orientation.

The relation between tendon length deviations (mean-removed) and the curvature components is linear:

$$\Delta l'_j = -r_t L\big(\kappa_x \cos\theta_j + \kappa_y \sin\theta_j\big), \qquad j = 1, \ldots, 4. \tag{4}$$

For the symmetric corner angles θ_j above this system admits a simple closed-form least-squares solution:

$$\begin{aligned} \kappa_x &= -\frac{1}{2r_t L}\sum_{j=1}^{4} \cos\theta_j \, \Delta l'_j \\ &= -\frac{\sqrt{2}}{4r_t L}\big(\Delta l'_1 - \Delta l'_2 - \Delta l'_3 + \Delta l'_4\big), \end{aligned} \tag{5}$$

$$\begin{aligned} \kappa_y &= -\frac{1}{2r_t L}\sum_{j=1}^{4} \sin\theta_j \, \Delta l'_j \\ &= -\frac{\sqrt{2}}{4r_t L}\big(\Delta l'_1 + \Delta l'_2 - \Delta l'_3 - \Delta l'_4\big) \end{aligned} \tag{6}$$

From these components we obtain

$$\kappa = \sqrt{\kappa_x^2 + \kappa_y^2}, \qquad \beta = \operatorname{atan2}(\kappa_y, \kappa_x), \qquad \phi = \kappa\, L. \tag{7}$$

B. Forward Kinematics

Given the curvature κ and bending-plane orientation β from the tendon length changes in Eq. (7), the tip pose of a segment relative to its base can be obtained via PCC modeling.

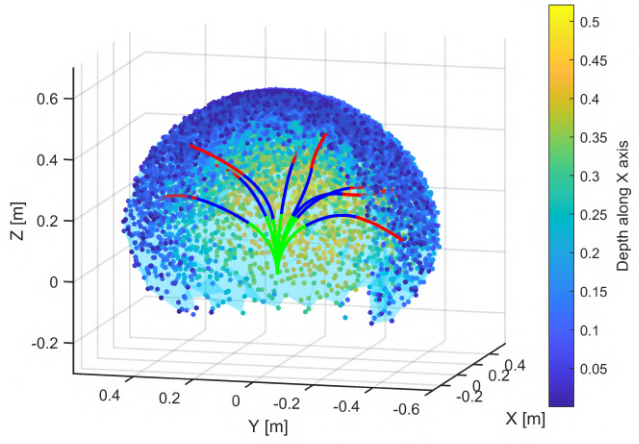

Fig. 2. Reachable workspace of the continuum robot (segment length 0.2m), nearly spherical for high dexterity.

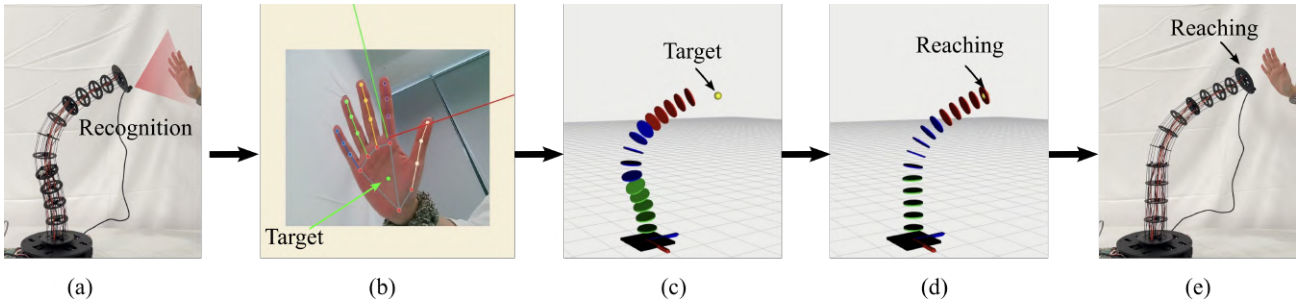

Fig. 3. Vision-guided palm detection and reaching process. (a) Palm recognition by the vision module. (b) Palm center extraction and target identification. (c) Projection of the target point into the robot coordinate frame. (d) Planned reaching trajectory in RViz. (e) Physical continuum manipulator reaching the target.

Assuming the segment bends as a circular arc of radius $\rho = 1/\kappa$ and central angle $\phi = \kappa L$, its tip position in the local frame $\mathcal{F}^{(0)}$—with z aligned along the undeformed backbone—is

$$\mathbf{p}^{(0)} = \begin{bmatrix} \rho(1-\cos\phi) \\ 0 \\ \rho\sin\phi \end{bmatrix}. \tag{8}$$

Here, the x-component represents the chordwise shift due to bending, and the z-component is the projection of the arc length onto the backbone axis. For a general bending-plane orientation β, a rotation about the z-axis gives

$$\mathbf{p} = \begin{bmatrix} \rho\cos\beta(1-\cos\phi) \\ \rho\sin\beta(1-\cos\phi) \\ \rho\sin\phi \end{bmatrix}. \tag{9}$$

The tip frame also rotates relative to the base. In PCC, this is captured by a rotation of angle ϕ about the local y-axis. Combining this with the rotation by β and the tip displacement yields the homogeneous transformation from segment base to tip:

$$\mathbf{T}_i^{i+1} = \begin{bmatrix} c_\beta c_\phi & -s_\beta & c_\beta s_\phi & c_\beta(1-c_\phi)/\kappa \\ s_\beta c_\phi & c_\beta & s_\beta s_\phi & s_\beta(1-c_\phi)/\kappa \\ -s_\phi & 0 & c_\phi & s_\phi/\kappa \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{10}$$

$$c_\beta = \cos\beta, \quad s_\beta = \sin\beta, \quad c_\phi = \cos\phi, \quad s_\phi = \sin\phi.$$

For a continuum manipulator composed of multiple PCC segments, the overall end-effector pose is obtained by sequentially multiplying the individual segment transformations from base to tip. To illustrate the reachable workspace resulting from this kinematic structure, Fig. 2 shows the tip positions of a three-segment continuum robot with segment length 0.2 m. The workspace is nearly spherical, demonstrating the high dexterity achievable with multiple PCC segments and the effectiveness of the PCC-based forward kinematics model in capturing the manipulator's configuration range.

III. Methodology

Achieving smooth and intention-consistent motion for continuum manipulators in vision-driven human–robot interaction scenarios demands a seamless integration of perception and planning. The user's intended pose is captured through a camera-based gesture recognition system, which provides a flexible and non-intrusive means to sense human motion in real time. Raw visual estimates, however, are inherently susceptible to sensor noise and the nonlinear dynamics of continuum manipulators; to mitigate these effects and maintain temporal coherence, the observations are filtered using an Extended Kalman Filter (EKF), ensuring that the planner receives stable and reliable guidance. Motion trajectories are then computed using a sliding-window trajectory optimization scheme, which balances the competing requirements of dynamic feasibility, curvature, and acceleration constraints, and responsiveness to the user's intent over short time horizons. By grounding trajectory planning in filtered visual intention and optimizing within a receding horizon, the framework effectively reconciles human intent with the mechanical limitations of the manipulator, enabling fluid and predictable motion in interactive environments.

A. Vision-Based Intent Trajectory EKF for Human-Robot Interaction

To enable smooth and responsive motion of the continuum manipulator that accurately reflects the user's intended hand movements, it is essential to obtain reliable and timely estimates of the user's target position. In order to achieve real-time perception without constraining the user's natural interaction, we employ a standard 1080P USB camera to capture RGB images of the user's hand. Hand detection and tracking is performed using Mediapipe to identify key landmarks, from which the palm center is extracted as the intended target. The depth of the hand is inferred by comparing the measured hand width with a nominal human hand width, yielding a 3D target position in the camera frame:

$$\hat{\mathbf{p}}_{\text{cam}}(t) = \begin{bmatrix} \hat{x}(t) \\ \hat{y}(t) \\ \hat{z}(t) \end{bmatrix}. \tag{11}$$

This position is transformed into the robot's world frame using the end-effector-to-world homogeneous transformation:

$$\hat{\mathbf{p}}_{\text{world}}(t) = \mathbf{T}_{\text{ee}}^{\text{world}}(t) \begin{bmatrix} \hat{\mathbf{p}}_{\text{cam}}(t) \\ 1 \end{bmatrix}. \tag{12}$$

Algorithm 1 Sliding-Window Trajectory Optimization with Intent Estimation

Require: Raw target trajectory $\hat{\mathbf{T}}_t$, initial joint guess $\mathbf{Q}_k^{(0)}$, weight matrices $\mathbf{W}_a, \mathbf{W}_p, \mathbf{W}_v$, tolerance ε, max iterations $j_{\max}$
Ensure: Optimized joint trajectory $\mathbf{Q}_k^*$
EKF Filtering: Initialize state $\mathbf{s}_0$, covariance $\mathbf{\Sigma}_0$
Propagate $\mathbf{s}_{t|t-1} = f(\mathbf{s}_{t-1|t-1}, \Delta t)$
Update $\mathbf{s}_{t|t} = \mathbf{s}_{t|t-1} + \mathbf{K}_t(\mathbf{y}_t - h(\mathbf{s}_{t|t-1}))$
Set filtered segment $\mathcal{P}_{\text{filtered}} = \{\mathbf{p}_{t|t}\}$
SQP Optimization Initialization: Compute $\nabla\mathcal{J}(\mathbf{Q}_k^{(0)})$, $\mathbf{c}(\mathbf{Q}_k^{(0)})$; $j \leftarrow 0$
while $\|\nabla\mathcal{J}(\mathbf{Q}_k^{(j)})\| > \varepsilon \ \wedge \ j < j_{\max}$ **do**
    Form Hessian $\mathbf{H}^{(j)} \approx \nabla^2\mathcal{L}(\mathbf{Q}_k^{(j)}, \boldsymbol{\lambda}^{(j)})$
    Linearize constraints: $\mathbf{c}(\mathbf{Q}_k^{(j)}) + \nabla\mathbf{c}^\top\delta\mathbf{Q} \le 0$
    Solve QP: $\min_{\delta\mathbf{Q}} \frac{1}{2}\delta\mathbf{Q}^\top\mathbf{H}^{(j)}\delta\mathbf{Q} + \nabla\mathcal{J}(\mathbf{Q}_k^{(j)})^\top\delta\mathbf{Q}$
    Update trajectory: $\mathbf{Q}_k^{(j+1)} \leftarrow \mathbf{Q}_k^{(j)} + \delta\mathbf{Q}$
    Update multipliers $\boldsymbol{\lambda}^{(j+1)}$ and gradient $\nabla\mathcal{J}(\mathbf{Q}_k^{(j+1)})$
    $j \leftarrow j + 1$
end while
return $\mathbf{Q}_k^* = \mathbf{Q}_k^{(j)}$

To provide smooth and temporally consistent trajectory estimates, we employ an Extended Kalman Filter (EKF) that tracks the user's intended motion. For each potential intent trajectory (i), the filter maintains position, orientation, and their velocities:

$$\mathbf{s}_t = \begin{bmatrix} \mathbf{p}_t^{(i)} & \mathbf{v}_t^{(i)} & \boldsymbol{\phi}_t^{(i)} & \boldsymbol{\omega}_t^{(i)} \end{bmatrix}^\top, \tag{13}$$

At each step, the EKF predicts the next state and incorporates measurements of position and orientation, producing filtered states $\mathbf{p}_{t|t}^{(i)}$ and $\boldsymbol{\phi}_{t|t}^{(i)}$ for downstream control.

These filtered targets are used in a sliding-window trajectory optimization to compute the joint configurations that guide the robot end-effector toward the intended position, as summarized in Algorithm 1. The full pipeline—from vision-based target detection, trajectory filtering, to robot execution—is illustrated in Fig. 3.

B. Sliding-Window Trajectory Optimization

To ensure that the continuum manipulator executes motions that are both dynamically feasible and responsive to the user's intended trajectory, it is necessary to consider a short horizon of upcoming intentions rather than relying solely on instantaneous targets. This motivates the use of a sliding-window trajectory optimization framework, which generates joint-space trajectories over a finite horizon while continuously incorporating the latest filtered intention estimates. Given a filtered intention sequence within a sliding window of length T_w:

$$\mathcal{P}_{\text{intent}} = \{\mathbf{p}_{\text{filtered}}(t_k), \mathbf{p}_{\text{filtered}}(t_{k+1}), \ldots, \mathbf{p}_{\text{filtered}}(t_{k+T_w})\} \tag{14}$$

where k denotes the current time index, we formulate the trajectory optimization as a constrained nonlinear program over the joint space trajectory segment:

$$\mathbf{Q}_k = \{\mathbf{q}_k, \mathbf{q}_{k+1}, \ldots, \mathbf{q}_{k+T_w}\}, \quad \mathbf{q}_i \in \mathbb{R}^n \tag{15}$$

1) Objective Function Formulation: The composite cost function integrates three fundamental objectives through a weighted combination that balances competing requirements:

$$\begin{aligned} \mathcal{J}(\mathbf{Q}_k) &= \mathcal{J}_{\text{smooth}} + \lambda_1\mathcal{J}_{\text{align}} + \lambda_2\mathcal{J}_{\text{cont}} \\ &= \sum_{i=k+1}^{k+T_w-1} \|\ddot{\mathbf{q}}_i\|_{\mathbf{W}_a}^2 + \lambda_1 \sum_{i=k}^{k+T_w} \|\mathbf{f}_{\text{FK}}(\mathbf{q}_i) - \mathbf{p}(t_i)\|_{\mathbf{W}_p}^2 \\ &\quad + \lambda_2 \sum_{i=k}^{k+T_w-1} \|\mathbf{q}_{i+1} - \mathbf{q}_i\|_{\mathbf{W}_v}^2 \end{aligned} \tag{16}$$

where $\ddot{\mathbf{q}}_i = \frac{\mathbf{q}_{i+1} - 2\mathbf{q}_i + \mathbf{q}_{i-1}}{\Delta t^2}$ represents the discrete joint acceleration. The three terms respectively address kinematic smoothness through acceleration penalization, intention alignment by minimizing end-effector tracking error in Cartesian space, and trajectory continuity via joint velocity regularization. Here, $\mathbf{f}_{\text{FK}} : \mathbb{R}^n \to \mathbb{R}^3$ denotes the forward kinematics mapping, and $\mathbf{W}_a, \mathbf{W}_p, \mathbf{W}_v \succ 0$ are positive definite weight matrices that allow for task-specific tuning of each objective component.

2) Constraint Formulation: The optimization framework incorporates multiple constraint categories to ensure feasible and safe robot operation:

$$\begin{cases} \kappa(\mathbf{q}_i) \le \kappa_{\max}, \quad \forall i \in [k, k+T_w] \\ \|\mathbf{f}_{\text{FK}}(\mathbf{q}_{k+T_w}) - \mathbf{p}_{\text{filtered}}(t_{k+T_w})\| \le \epsilon_{\text{terminal}} \end{cases} \tag{17}$$

3) Optimization Framework: The complete trajectory optimization problem is formulated as a nonlinear program that simultaneously considers all objectives and constraints:

$$\begin{aligned} \mathbf{Q}_k^* = \arg\min_{\mathbf{Q}_k} \quad & \mathcal{J}(\mathbf{Q}_k) \\ \text{s.t.} \quad & (17) \end{aligned} \tag{18}$$

To solve this nonlinear optimization problem efficiently, we employ a SQP approach with Gauss-Newton Hessian approximation. At each iteration j, the algorithm solves a quadratic subproblem that linearizes the constraints and approximates the objective function:

$$\begin{aligned} \min_{\delta\mathbf{Q}} \quad & \frac{1}{2}\delta\mathbf{Q}^\top\mathbf{H}^{(j)}\delta\mathbf{Q} + \nabla\mathcal{J}(\mathbf{Q}^{(j)})^\top\delta\mathbf{Q} \\ \text{s.t.} \quad & \nabla\mathbf{c}(\mathbf{Q}^{(j)})^\top\delta\mathbf{Q} + \mathbf{c}(\mathbf{Q}^{(j)}) \le 0 \end{aligned} \tag{19}$$

where $\mathbf{H}^{(j)} = \nabla^2\mathcal{L}(\mathbf{Q}^{(j)}, \boldsymbol{\lambda}^{(j)})$ represents the Hessian of the Lagrangian, and $\mathbf{c}$ aggregates all constraint functions.

Given the complex nature of curvature calculations for continuum robots, we implement a smooth approximation that facilitates gradient-based optimization:

$$\kappa(\mathbf{q}) \approx \|\mathbf{J}_\kappa(\mathbf{q})\dot{\mathbf{q}}\|_2, \quad \mathbf{J}_\kappa = \frac{\partial\kappa}{\partial\mathbf{q}} \tag{20}$$

where $\mathbf{J}_\kappa$ is the curvature Jacobian derived from continuum robot kinematics.

The overall workflow of the algorithm is illustrated in Algorithm 1, where the EKF is used to continuously refine the intended trajectory, providing smooth and temporally

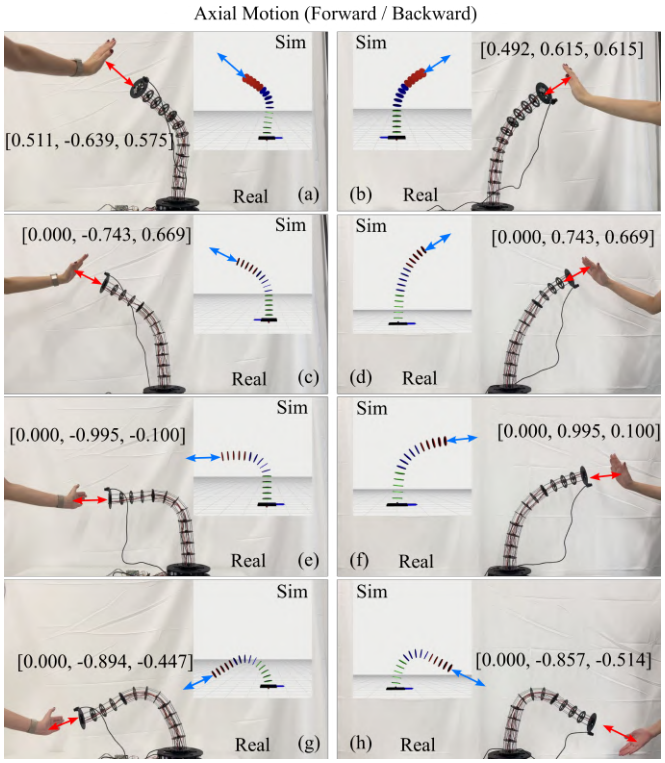

Fig. 4. Eight-direction following experiments. The continuum robot was commanded to follow palm motions in (a-h) eight orientations.

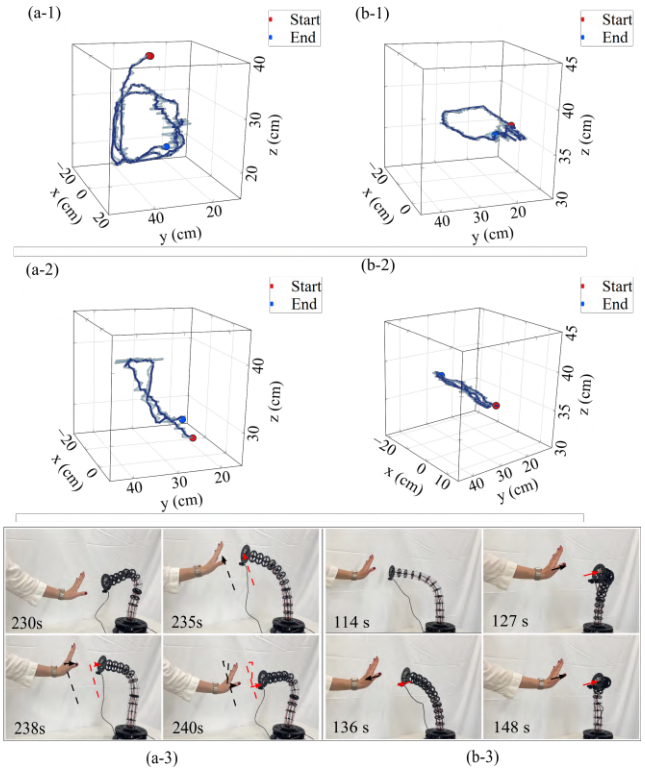

Fig. 5. (a-1) 3D trajectory from experiment in figure 6 (a-1) showing raw and filtered data, (a-2) another yz-plane trajectory with different shape, (a-3) corresponding experiment screenshots of (a-2); (b-1) 3D trajectory from experiment in figure 6 (b-1), (b-2) xy-plane trajectory, (b-3) corresponding experiment screenshots of (b-2).

consistent estimates that feed into the SQP-based iterative optimization of the joint trajectory. This tightly coupled process ensures that the robot follows the human's intended motion while satisfying kinematic constraints.

IV. Experiments and Results

To validate the proposed framework, we conducted a series of experiments involving palm detection, target generation, and multi-directional following within the reachable workspace. All experiments were performed both in simulation and on the physical continuum robot platform.

A. Palm Detection and Target Generation

The first set of experiments focused on obtaining reliable palm detection results and transforming the detected palm center into the robot's coordinate frame. Fig. 3 shows representative image frames where the palm region was identified and the palm center was marked with a green dot in (b). Despite variations in hand orientation and lighting conditions, the vision module consistently extracted stable palm features in real time.

Once the palm center was detected in the image plane, it was projected into 3D space through camera calibration and coordinate transformation. The resulting target point was visualized in the RViz environment, as illustrated in Fig. 3(c). In this figure, the yellow sphere represents the detected interaction target, and the continuum manipulator is rendered in its current configuration. This visualization confirms that the vision-based detection and coordinate transformation pipeline produced spatially consistent targets suitable for trajectory optimization.

B. Multi-Directional Following

Based on this workspace analysis in Fig. 2, eight representative orientations were selected for testing. In each case, the robot's end-effector camera was oriented toward the designated direction, while the human palm was placed within the field of view and moved forward and backward along the depth axis.

During these trials, the proposed framework continuously detected the palm center, transformed it into the robot's coordinate frame, and optimized trajectories in real time. Fig. 4 presents the results of the eight-direction experiments in both real world and Rviz simulation. Across all orientations, the continuum robot exhibited smooth and continuous tracking of palm motions, with no noticeable latency or discontinuity. This demonstrates the robustness of the framework in handling diverse interaction scenarios.

C. Forward-Oriented Interaction Experiments

To evaluate the continuum manipulator under forward-facing scenarios with complex palm motions, two representative trajectories were tested.

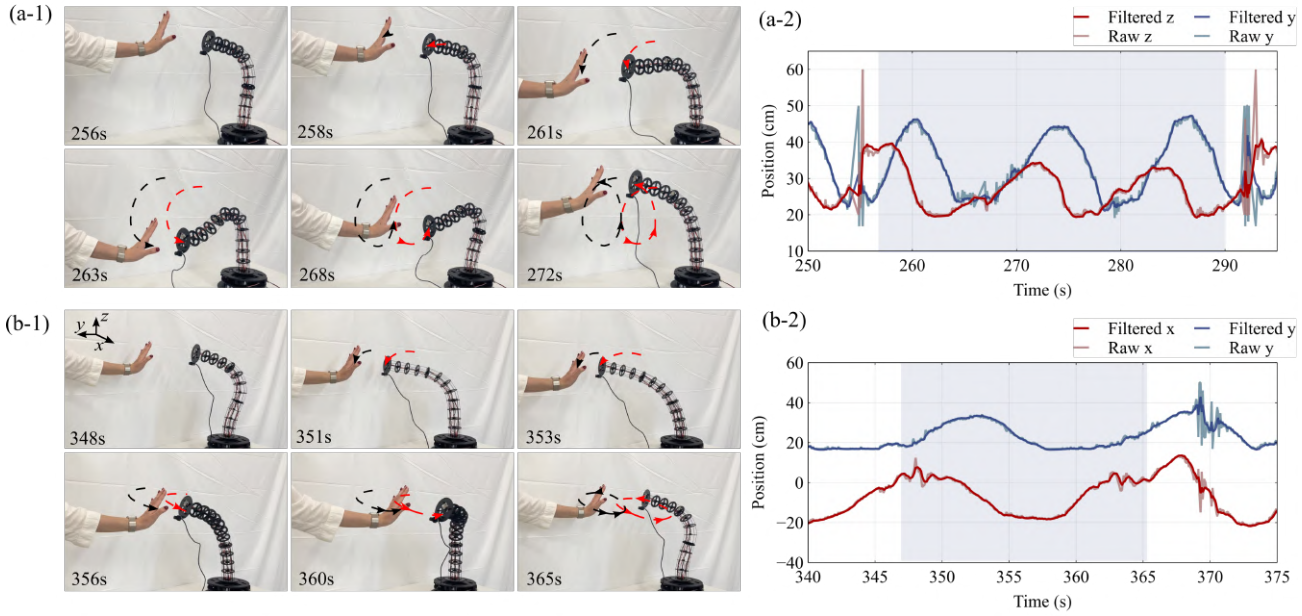

Fig. 6. Representative results of palm-guided interaction experiments. (a-1) Snapshots from experiment (a) with circular palm motion in the yz-plane. (a-2) Corresponding raw (dashed) and EKF-filtered (solid) trajectories. (b-1) Snapshots from experiment (b) with circular palm motion in the xy-plane. (b-2) Corresponding raw and filtered trajectories. In both cases, EKF filtering produces smoother and more stable motions, effectively suppressing visual noise and jitters.

TABLE I
VELOCITY STATISTICS OF TARGET TRAJECTORIES BEFORE AND AFTER FILTERING (YZ PLANE)

	Raw	Filtered	Δ	Reduction (%)
Y Direction				
Avg Velocity	0.743	0.302	0.441	59.3
Velocity Std	2.054	0.550	1.504	73.2
Max Velocity Change	94.776	16.888	77.887	82.2
Z Direction				
Avg Velocity	0.449	0.224	0.225	50.2
Velocity Std	1.414	0.408	1.006	71.2
Max Velocity Change	53.925	11.293	42.632	79.1

Experiment (a) involved a palm tracing an approximately circular path in the yz-plane to assess periodic and closed-loop motion. The raw trajectories exhibit fluctuations due to camera noise and depth estimation, while EKF filtering produces significantly smoother trajectories that better capture the underlying motion trends. Experiment (b) examined circular palm motion in the xy-plane for horizontal movements, showing similar smoothing effects.

Analysis of trajectory segments confirms that the manipulator maintains smooth, continuous motion. EKF-filtered trajectories effectively suppress high-frequency noise without distorting the intended motion, enabling the robot to reliably reproduce both linear and closed-loop trajectories with consistent responses. The circular trajectory further demonstrates the framework's ability to handle continuous, closed motions without deviation.

Table I quantitatively demonstrates the effectiveness of EKF filtering in stabilizing target trajectories in the YZ plane. In the Y direction, the maximum velocity change decreases dramatically from 94.8m/s to 16.9m/s, reflecting a significant reduction in abrupt fluctuations. Meanwhile, in the Z direction, the standard deviation drops from 1.41m/s to 0.41m/s, indicating that the overall motion becomes much smoother and more consistent. Along with reductions in average velocities, these improvements show that the filtered trajectories are substantially more stable and reliable than the raw measurements. Such stability is critical for providing accurate motion references to the robot, ensuring smooth tracking and safe human–robot interaction.

Overall, the experiments confirm that EKF filtering is essential for mitigating vision noise and stabilizing target trajectories, while also demonstrating the robustness and effectiveness of our framework in executing complex, nonlinear human–robot interaction tasks.

D. Up-Oriented Interaction Experiments

To evaluate the effectiveness of our motion planner, we conducted experiments with the proposed sliding-window trajectory optimization framework. Within each time window T_w, the filtered palm intention sequence $\mathcal{P}_{\text{intent}}$ (Eq. (14)) was used as the reference, while the corresponding joint-space trajectory $\mathbf{Q}_k$ (Eq. (15)) was optimized according to the composite cost function in Eq. (16). This ensured that the generated trajectories not only followed the human intention but also maintained smoothness and continuity.

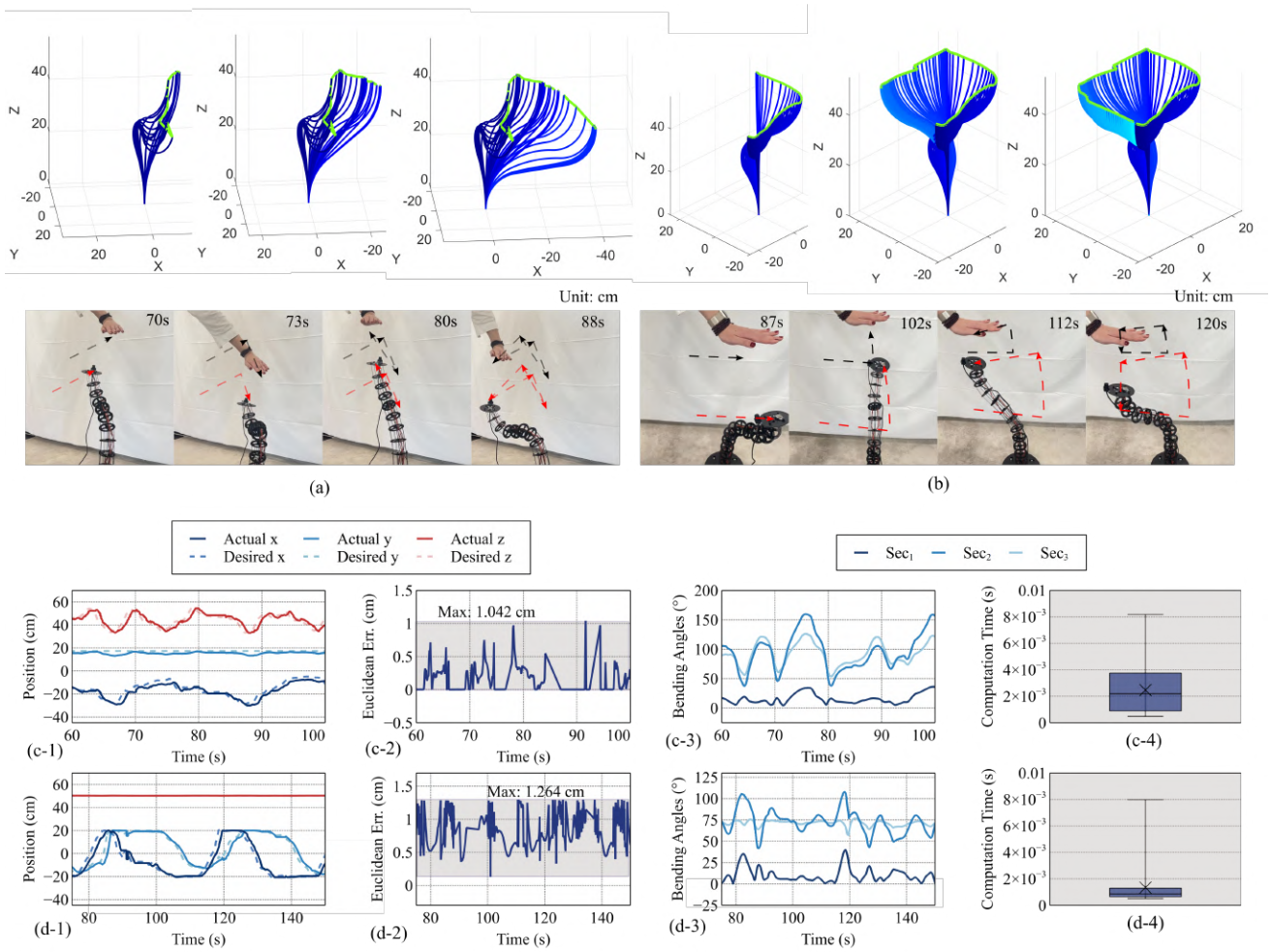

Fig. 7. Trajectory tracking of the continuum robot in different planes. (a) End-effector motion sequence in the vertical plane (steps 1–6). (b) Motion sequence in the horizontal plane (steps 1–6). (c-1) Position tracking in the vertical plane (actual vs. desired). (c-2) Euclidean error. (c-3) Joint angle variations (θ_1, θ_2, θ_3). (c-4) Trajectory planning time per IK step. (d-1)–(d-4) Corresponding results for the horizontal plane, presented in the same format as (c-1)–(c-4).

TABLE II
COMPARISON WITH OTHER WORKS.

Structure (L)	Planning Time (ms per step)	Pos. Error (cm)	Rel. Error (%)	Visual Guidance	Avg. Angular Vel. (deg/s)
Ours (L = 600 mm)	2.5	1.042	<1.8	Yes	5.22
Soft Microcatheters (L = 70 mm) [24]	6.5	1.3	<2.5	No	–
Hyperelastic continuum robots (L = 90 mm) [25]	77	<4.5	<5	No	–
Soft manipulator (L = 536 mm) [26]	8.2	<13.4	<2.5	No	–

Fig. 7 illustrates the deformation behavior and tracking performance of the continuum manipulator. Subplots (a) and (b) show that bending is smoothly distributed along multiple segments in both vertical and horizontal planes, validated in simulation and real experiments. Subplots (c-1)/(d-1) compare actual versus desired end-effector positions, with Euclidean errors (c-2)/(d-2) remaining low—maximum deviations are 1.042 cm in the vertical plane and 1.264 cm in the horizontal plane. The joint bending angles $\theta_1, \theta_2, \theta_3$ (c-3)/(d-3) vary continuously without abrupt changes, demonstrating global coordination, while computation times (c-4)/(d-4) remain within real-time requirements.

In addition, Table II compares our framework with representative prior works. Our method achieves higher position accuracy and faster planning time while also benefiting from visual guidance. More importantly, we uniquely report an angular velocity metric, which highlights the smoothness and continuity of the generated motions—an aspect not evaluated in previous studies. Together, these results confirm that the proposed optimization framework ensures accurate, real-time,

and smooth trajectory execution, making it well-suited for robust hand-intention following in complex tasks.

V. Conclusion and Future Work

In this work, we developed a real-time framework that estimates short-term human target poses from vision and generates smooth, curvature-feasible trajectories for tendon-driven continuum robots. The framework employs Extended Kalman Filter (EKF) for pose estimation and a sliding-window trajectory optimizer to ensure responsive and accurate execution. A physical prototype was built, and extensive real-world experiments demonstrated that the proposed approach achieves smooth robot deformation, accurate trajectory tracking, and low-latency performance in dynamic human–robot interaction scenarios. These results confirm the feasibility and practical effectiveness of our method in enabling safe, intuitive, and responsive collaboration.

Future work will extend intention estimation beyond palm tracking by incorporating multimodal sensing, and further improve trajectory planning through dynamic obstacle avoidance and more efficient optimization algorithms.

References

[1] R. J. Webster and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[2] R. Peng, Y. Wang, M. Lu, and P. Lu, "A dexterous and compliant aerial continuum manipulator for cluttered and constrained environments," *Nature Communications*, vol. 16, no. 1, p. 889, 2025.

[3] I. D. Walker, D. M. Dawson, T. Flash, F. W. Grasso, R. T. Hanlon, B. Hochner, W. M. Kier, C. C. Pagano, C. D. Rahn, and Q. M. Zhang, "Continuum robot arms inspired by cephalopods," in *Unmanned Ground Vehicle Technology VII*, ser. Proceedings of SPIE, vol. 5804. SPIE, 2005, pp. 303–314.

[4] J. Zhang, Q. Fang, P. Xiang, D. Sun, Y. Xue, R. Jin, K. Qiu, R. Xiong, Y. Wang, and H. Lu, "A survey on design, actuation, modeling, and control of continuum robot," *Cyborg and Bionic Systems*, vol. 2022, p. 9754697, 2022.

[5] K. Shahzad, S. Iqbal, and P. Bloodsworth, "Points-based safe path planning of continuum robots," *International Journal of Advanced Robotic Systems*, vol. 12, no. 7, 2015.

[6] B. H. Meng, I. S. Godage, and I. Kanj, "Rrt*-based path planning for continuum arms," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6830–6837, 2022.

[7] A. Ataka, P. Qi, H. Liu, and K. Althoefer, "Real-time planner for multi-segment continuum manipulator in dynamic environments," in *Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4080–4085.

[8] Y. Wen and P. Pagilla, "Path-constrained and collision-free optimal trajectory planning for robot manipulators," *IEEE Transactions on Automation Science and Engineering*, vol. 20, no. 2, pp. 763–774, Apr 2023.

[9] J. Chen, J. Yan, Y. Qiu, H. Fang, J. Chen, and S. S. Cheng, "A cross-entropy motion planning framework for hybrid continuum robots," *IEEE Robotics and Automation Letters*, vol. 8, no. 12, pp. 8200–8207, 2023.

[10] M. Hammond, V. Cichella, A. F. Golestaneh, and C. Lamuta, "Path planning for continuum rods using bernstein surfaces," in *2024 IEEE 7th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2024, pp. 109–114.

[11] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2021.

[12] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 7745–7754.

[13] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3d convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 1–7.

[14] P. Molchanov, S. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4207–4215, 2016.

[15] N. Neverova, C. Wolf, G. Taylor, and F. Nebout, "Moddrop: Adaptive multi-modal gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1692–1706, 2016.

[16] B. Xu, J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli, "Interact as you intend: Intention-driven human-object interaction detection," *IEEE Transactions on Multimedia*, vol. 22, no. 6, pp. 1423–1432, 2020.

[17] N. Ratliff, J. Zucker, J. A. Bagnell, and S. Srinivasa, "Chomp: Gradient optimization techniques for efficient motion planning," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2009, pp. 489–494.

[18] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding locally optimal, collision-free trajectories with sequential convex optimization," in *Robotics: Science and Systems*, vol. 9, 2013, pp. 1–10.

[19] M. Neunert *et al.*, "Fast nonlinear model predictive control for unified trajectory optimization and tracking," in *Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, 2016, pp. 1398–1404.

[20] C. Ji, Z. Zhang, G. Cheng, M. Kong, and R. Li, "A convex optimization method to time-optimal trajectory planning with jerk constraint for industrial robotic manipulators," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 4, pp. 7629–7646, Oct 2024.

[21] A. Palleschi *et al.*, "Time-optimal trajectory planning for flexible joint robots," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 938–945, Apr 2020.

[22] R. Peng, P. Deng, D. Tang, and P. Lu, "Numerical optimization-based kinematics with pose tracking control for continuum robots," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 8416–8423.

[23] R. Peng, Y. Wang, and P. Lu, "A tendon-driven continuum manipulator with robust shape estimation by multiple imus," *IEEE Robotics and Automation Letters*, vol. 9, no. 4, pp. 3084–3091, 2024.

[24] D. Lin, W. Chen, K. He, N. Jiao, Z. Wang, and L. Liu, "Position and orientation control of multisection magnetic soft microcatheters," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 2, pp. 907–918, 2023.

[25] J. Shi, S.-A. Abad, J. S. Dai, and H. A. Wurdemann, "Position and orientation control for hyperelastic multisegment continuum robots," *IEEE/ASME Transactions on Mechatronics*, vol. 29, no. 2, pp. 995–1006, 2024.

[26] Z. Gong, X. Fang, X. Chen, J. Cheng, Z. Xie, J. Liu, B. Chen, H. Yang, S. Kong, Y. Hao *et al.*, "A soft manipulator for efficient delicate grasping in shallow water: Modeling, control, and real-world experiments," *The International Journal of Robotics Research*, vol. 40, no. 1, pp. 449–469, 2021.