IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Hysteresis-Aware Neural Network Modeling and Whole-Body Reinforcement Learning Control of Soft Robots

Zongyuan Chen, Yan Xia, Jiayuan Liu, Jijia Liu, Wenhao Tang, Jiayu Chen, Feng Gao, Longfei Ma, Hongen Liao, Yu Wang, Chao Yu, Boyu Zhang, Fei Xing

***Abstract*—Soft robots are inherently compliant and safe, making them suitable for human-interactive applications such as surgery. However, their nonlinear and hysteretic behavior, arising from the properties of soft materials, presents substantial challenges for accurate modeling and control. In this study, we present a soft robotic system and propose a hysteresis-aware whole-body neural network model that accurately captures and predicts the soft robot's whole-body motion, including its hysteretic behavior. Building upon the high-precision dynamic model, we construct a highly parallel simulation environment for soft robot control and apply an on-policy reinforcement learning algorithm to efficiently train whole-body motion control policies. The trained policy is deployed on the real soft robot to evaluate its control performance. Furthermore, we develop a soft robotic system for surgical applications and validate it through phantom-based laser ablation experiments. The results demonstrate that the hysteresis-aware modeling reduces the Mean Squared Error (MSE) by 86.07% compared with traditional modeling methods. The deployed control algorithm achieves a trajectory tracking error ranging from 0.147 to 0.307 mm on the real soft robot, highlighting its precision in real-world conditions. The proposed method also shows strong performance in phantom-based surgical experiments, and demonstrates its potential for complex scenarios, including future real-world clinical applications.**

***Index Terms*—Soft robotics, reinforcement learning, robot control, dynamics modeling**

I. Introduction

SOFT robots are typically constructed from soft material that extends, bends, and twists according to the actuation provided by air pressure or cables [1]. They have a continuously deformable structure that offers enhanced flexibility, compliance, and a high degree of freedom [2], [3], [4]. In recent years, soft pneumatic robots have been widely introduced into tasks involving interaction with the human body, especially in medical and surgical applications [5]. Compared with conventional rigid surgical instruments, soft robots can provide more degrees of freedom to navigate easily through the complex and tortuous structures in the abdominal cavity. They also exhibit good biocompatibility with tissues, lowering the risk of damage [6].

However, the dynamics of soft robots is highly influenced by their material and structural design. Modeling soft robots presents significant challenges due to the intrinsic material nonlinearity [7] and hysteresis properties. The difficulty in obtaining accurate models for precise control increases the risk of failure or damage in applications. Therefore, achieving accurate modeling and control of soft robots remains a critical and challenging research problem.

To model soft robots, conventional methods are mainly based on Piecewise Constant-Curvature (PCC) kinematic models [8] and Cosserat rod theory [9]. These methods are widely used for modeling soft robots with simple materials and structures. However, these methods rely on several simplifying assumptions, such as material homogeneity, constant circumferential geometry of pressure chambers during actuation, and the independence of multiple chambers. These assumptions are rarely met in practical soft robotic systems. As application scenarios broaden, soft-bodied robots are increasingly integrated with functional attachments, such as endoscopes or surgical tools in surgery. Such additions introduce non-uniform and complex structures that result in dynamic behaviors beyond the modeling capacity of conventional analytical methods. In recent years, data-driven methodologies have been applied for modeling flexible robots [10], [11]. Specifically, multi-layer perceptrons (MLPs) have demonstrated superior performance, attributable to their capacity for non-linear fitting and feature characterization [12].

The motion of pneumatic soft robots exhibits hysteresis, as their shape is influenced not only by the pressure magnitude but also by the direction of pressure change. As a result, identical pressure inputs can produce different outputs during pressurization and depressurization, becoming a primary source of modeling error [13]. Previous studies have used recurrent neural networks (RNNs) to process sequential data and convolutional neural networks (CNNs) to model spatially rearranged pressure inputs. For example, [14] modeled hysteresis

Manuscript received April 18, 2025; Revised: August 9, 2025; Accepted: August 31, 2025. This paper was recommended for publication by Editor Cecilia Laschi upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Grant U24A6006, Grant 62406159, Grant 62325405, Grant 62403307, in part by the Postdoctoral Fellowship Program of CPSF under Grant GZC20240830 and Grant 2024M761676, in part by the China Postdoctoral Science Special Foundation under Grant 2024T170496, and in part by the Science and Technology Commission of Shanghai Municipality under Grant 24ZR1439800. (Zongyuan Chen and Yan Xia contributed equally to this work.) (Corresponding authors: Chao Yu; Boyu Zhang; Fei Xing.)

Zongyuan Chen, Yan Xia, Jiayuan Liu, Jijia Liu, Wenhao Tang, Jiayu Chen, Feng Gao, Longfei Ma, Yu Wang, Fei Xing are with Tsinghua University, Beijing 100084, China (e-mail: xingfei@mail.tsinghua.edu.cn). Chao Yu is with Tsinghua University, Beijing 100084, China, and also with Beijing Zhongguancun Academy, Beijing 100094, China (e-mail: zoeyuchao@gmail.com).

Hongen Liao is with Tsinghua University, Beijing 100084, China, and also with Shanghai Jiao Tong University, Shanghai, 200030, China. Boyu Zhang is with Shanghai Jiao Tong University, Shanghai, 200030, China (e-mail: boyuzhang@sjtu.edu.cn).

Digital Object Identifier (DOI): see top of this page.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

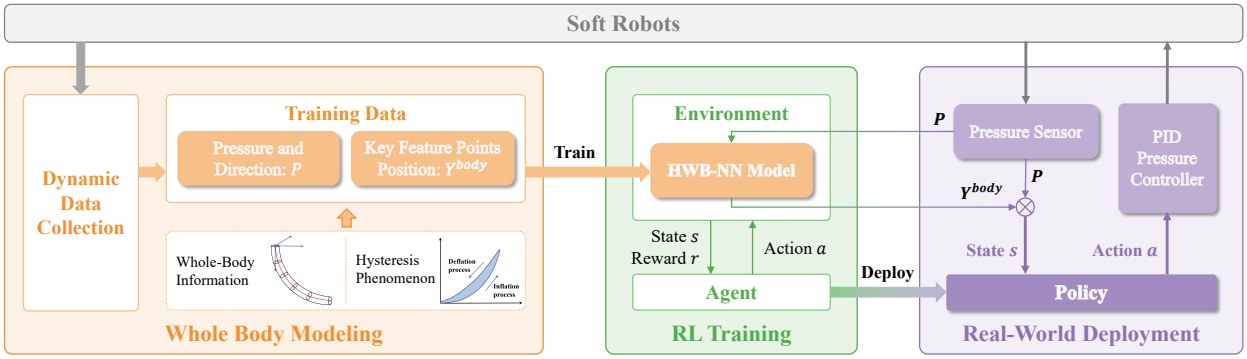


Fig. 1: The proposed framework combining hysteresis-aware whole-body modeling and reinforcement learning for soft robot control.

by inferring the execution sequence from the spatial pose. However, due to their complex structure, these models require large amounts of temporally continuous data [8]. In addition, most data-driven approaches focus solely on modeling the soft robot's end-effector, which is only sufficient for simple tasks. Meanwhile, in complex tasks such as surgery, the soft robot is expected to avoid obstacles and ensure safe interaction with the environment, which necessitates modeling its entire body.

Soft robots are underactuated systems characterized by a complex mapping between input pressures and resulting morphology, making the design of control algorithms particularly challenging [15]. Conventional control theory struggles to address whole-body control tasks for soft robots due to the high dimensionality of the state space. As an effective alternative, reinforcement learning methods have been proposed to directly learn control policies, demonstrated promising results in soft robot control [16], [17], [18]. For example, deep Q-networks (DQN) [19] have been used to develop open-loop position control strategies for pneumatic continuum robots, achieving target point tracking under quasi-static conditions. Other studies have combined trajectory optimization with supervised learning to derive closed-loop control policies [20]. However, in real-world complex tasks such as surgery, soft robots are required to operate in black-box environments while ensuring safe interaction with surrounding tissues. This necessitates accurate whole-body modeling and control that comprehensively addresses the aforementioned challenges.

In this paper, we design a three-chamber pneumatic soft robot system for surgical applications. We propose a novel intelligent control framework for soft robot: the hysteresis-aware neural network modeling and whole-body reinforcement learning algorithm.

The main contributions of this paper can be summarized as follows:

- We propose a Hysteresis-aware Whole-Body Neural Network (HWB-NN) that incorporates pneumatic actuation direction to model hysteresis and accurately predict the whole-body position of the soft robot, especially in scenarios where conventional modeling performs poorly, such as robots with non-uniform structures.
- Based on the HWB-NN model, we construct a highly parallel simulation environment for soft robot control and apply an on-policy reinforcement learning (RL) algorithm to efficiently train whole-body motion control policies. This approach provides a promising solution for complex operational tasks.
- A soft robotic system for laparoscopic surgery is developed. The proposed modeling and control framework is implemented and validated on the physical platform through trajectory tracking and phantom-based surgical tasks, demonstrating robustness and real-world applicability.

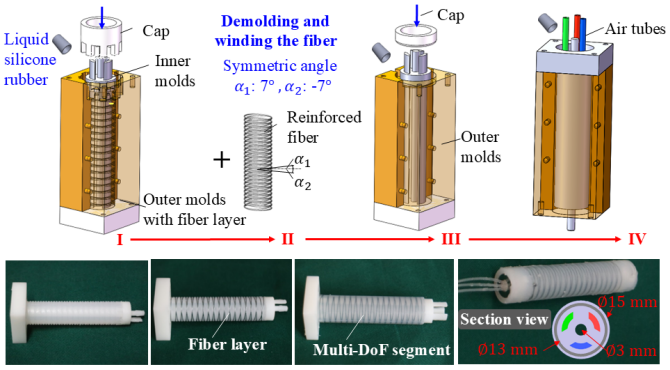


Fig. 2: Manufacturing process of the pneumatic soft robot.

The overall workflow of the proposed modeling and control framework is illustrated in Fig. 1.

II. Design and Fabrication of Soft Robots

To enhance the performance of the robot, we have designed a fiber-reinforced multidegree-of-freedom soft actuator, which effectively reduces the radial expansion of the actuator. The soft robot has a length of 70 mm and a diameter of 15 mm, with a central 3 mm channel reserved for tools such as endoscopes or laser devices. The air chambers are equally spaced at 120° along the cylinder.

The fabrication of the robot mainly involves five main steps. First, assemble the molds and spray the release agent. Pour in liquid silicone and cure to create spiral pattern segments. Second, wrap the reinforcing fibers around the silicone in a spiral pattern with a symmetrical angle of 7 °. This angle was selected based on the findings of Connolly et al. [21], which

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

show that small fiber angles (0°–20°) maximize axial extension. We chose 7° to further balance fabrication feasibility and motion stability. Third, after assembling the outer mold, pour liquid silicone to encapsulate the reinforcing fibers. Fourth, cast the end segment and attach the air tubes, as shown in Fig. 2.

III. Hysteresis-aware whole-body neural network dynamic modeling

Modeling soft robots presents several key challenges. Conventional analytical methods are limited in handling material inhomogeneity and integrated attachments. Data-driven approaches often fail to capture whole-body morphological information. Moreover, hysteresis effects induced by direction of actuation pressure can result in significant positioning errors. Specifically, when the same pressure is achieved through the pressurization and depressurization paths, the position of the end effector may differ significantly. Our experiments on the three-chamber soft robot revealed that this hysteresis follows complex and often overlooked patterns. We applied pressure to a single chamber from 0 kPa to 70 kPa in 5 kPa increments, holding each pressure level for 5 seconds to ensure stabilization, and then released the pressure back to 0 kPa with the same steps. The results showed that the average Euclidean deviation of the end-effector positions under identical pressures during pressurization and depressurization was 1.8 mm, with a maximum deviation of 2.4 mm, corresponding to 2.5% and 3.4% of the robot's total length, respectively.

To address these challenges, this paper proposes a hysteresis-aware neural network-based model that incorporates the hysteresis effect and captures the whole-body configuration of the soft robot. Trained on real-world data, the model does not rely on predefined actuator structures or material-specific mechanics, thereby enabling accurate dynamic modeling of soft robots with complex structures.

A. Hysteresis-aware Whole-Body Neural Network Model Design

The dynamic model of a soft robot describes the mapping between the input air pressure and the morphological configuration of the robot. The Piecewise Constant Curvature (PCC) model [7] assumes that a continuum robot consists of a series of piecewise continuous circular arcs. Inspired by the PCC model, we propose a hysteresis-aware piecewise key feature points neural network model. Several circular cross-sections perpendicular to the central axis of the soft robot are defined as key feature planes, and the center of each plane is designated as a key feature point $O_i(x_i, y_i, z_i)$. The soft robot is represented as a series of connected segments, with these key feature points marking the boundaries between adjacent segments. A neural network is used to predict the positions of the key feature points, and a B-spline curve fitting method is then applied to reconstruct the continuous morphology between them. Compared with vision-based approaches [22], the proposed piecewise modeling method based on point data is lower-dimensional, which significantly reduces the complexity of morphological data acquisition and simulation cost in 3D space, while providing a complete whole-body morphology representation. It also allows for adjustable complexity in morphology data acquisition and simulation cost based on task requirements, by varying the number of key feature points.

To address the strong nonlinearity and hysteresis in the relationship between actuation pressure and position, we propose a hysteresis-aware multilayer perceptron (MLP) architecture that incorporates both pressure magnitude and directional change as inputs. The specific network structure is as follows:

- Input layer: The input layer consists of 6 neurons, corresponding to the input air pressure values and their directions of change for the soft robot with 3 air chambers, and the input can be defined as:

$$P = [p_1, p_2, p_3, d_1, d_2, d_3] \in \mathbb{R}^6. \tag{1}$$

- Hidden layer: We experimentally evaluated the impact of different neural network architectures, varying both the number of hidden layers and the number of neurons per layer. The results of the comparative experiment are shown in Fig. 4. The model with the lowest test loss was selected, which consisted of 3 hidden layers, each containing 256 neurons. The ReLU activation function was applied after each hidden layer.
- Output layer: The output layer consists of $3 \times n$ neurons corresponding to 3-dimensional coordinates of n feature segmentation points, each of which can be represented as $O_i = (x_i, y_i, z_i)$, so the output Y is defined as:

$$\begin{aligned} \mathbf{Y} &= [\mathbf{O}_1, ..., \mathbf{O}_n] \in \mathbb{R}^{3\times n} \\ &= [O_{1x}, O_{1y}, ..., O_{nx}, O_{ny}, O_{nz}]. \end{aligned} \tag{2}$$

- Loss function: The fixed and free ends of soft robots have different motion ranges that differ by several orders of magnitude. The naive MSE loss may introduce instability in model training. We proposed a weighted MSE based on the different motion ranges of feature points. The motion range is defined as:

$$\Delta Y_i = max(Y_i) - min(Y_i). \tag{3}$$

The corresponding weights for each dimension output variable are defined as:

$$w_i = 1 + \frac{n \times \Delta Y_i}{\sum_{i=1}^{n} \Delta Y_i}. \tag{4}$$

The loss function weighted according to the range of motion is defined as:

$$Loss(Y, \hat{Y}) = \frac{1}{n} \sum_{i=1}^{n} w_i \times (Y_i - \hat{Y}_i)^2. \tag{5}$$

The model is evaluated on the validation set every 10 epochs, with early stopping applied if no improvement occurs for 10 consecutive evaluations to prevent overfitting.

B. Experimental Platforms Setup and Data Collection

A schematic of the data acquisition system is shown in Fig. 3. The soft robot is fixed on the experimental table and driven by a pneumatic system consisting of a set of syringe pumps. Each air chamber of the soft robot is driven by an

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

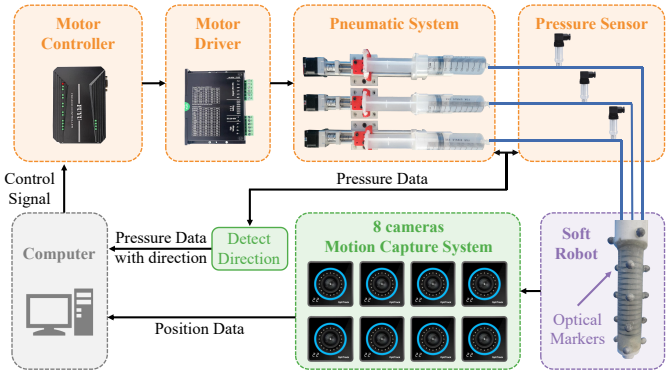


Fig. 3: Schematic of the data collection system designed to capture directional pressure inputs and whole-body motion data of the soft robot.

individual syringe pump. The piston position of the syringe pump is controlled by a lead screw slide table driven by a stepper motor, and the air pressure in the chamber is controlled by changing the volume in the syringe pump.

A pneumatic pressure sensor measures the system pressure in real time and transmits the measurements to a proportional–integral–derivative (PID) controller. The controller adjusts the motor speed to regulate the air pressure within the chamber. The soft robot generates different deformation motions according to the air pressure. Under step input conditions, the PID-controlled system achieves an average final error of 0.03 kPa and an average settling time of 0.794 s, based on 10 repeated trials.

The experimental table of the soft robot is surrounded by an infrared optical motion capture system composed of eight OptiTrack cameras that collect millimeter-level position data. The cameras are positioned in different directions to ensure each marker is visible to at least three cameras across the entire workspace, thereby mitigating occlusion caused by self-deformation or interaction.

We selected n planes perpendicular to the direction of the soft robot axis as the key feature planes. To achieve optimal modeling performance, n should be chosen as the maximum number of planes that can be placed without marker interference. In our experiments, this value was determined to be 5.

To ensure robust capture of the deformation of soft robots during bending and elongation, at least 3 infrared reflective markers are affixed to each key feature plane. The positional data of the markers are processed using a circle fitting algorithm to determine the center of each piecewise key feature plane, which is designated as key feature points. To minimize the influence of transverse expansion, the markers should be placed between air chambers, rather than being affixed to their surfaces. The first circle marker is attached to the fixed bracket at the base of the soft robot to obtain the initial point coordinates.

To collect a realistic dataset for dynamic modeling of the soft robot, the pressure in each air chamber is sequentially varied to traverse the entire action space at defined step intervals. Data collection is conducted under quasi-static conditions. After applying each set of target pressures, a 3-second delay is introduced to allow the soft robot to reach a stable state. Subsequently, 1 second of data is recorded, including air pressure readings from pressure sensors and positional data from markers located on the surface of the soft robot, captured by the motion capture system. Data are collected at a frequency of 20 Hz, and the mean value of each 1-second data segment is computed to represent a single valid data point. The input of the dataset consists of three-dimensional pressure values and their corresponding directional changes in each dimension. The directions of pressure increase and decrease are represented by 1 and -1, respectively. The output includes the spatial coordinates of the piecewise key feature points. A total of 13824 samples are collected as the training set. Additionally, validation and test sets containing 1000 samples each are collected separately, with distributions different from the training set.

IV. Reinforcement Learning Whole-Body Control for Soft Robots

To address the challenge of high-dimensional state space in soft robot whole-body control, handle complex task constraints, and bridge the sim-to-real gap, we propose a reinforcement learning method based on a parallel simulation environment built on the HWB-NN model. We formulate the control process as a Markov Decision Process (MDP) [23]. The policy is then trained using the Proximal Policy Optimization (PPO) [24] algorithm. The implementation details are described below.

A. MDP-Based Modeling of the Robot System

To apply reinforcement learning within a soft robot system, the motion process of the soft robot is represented as an MDP to capture the interaction between the robot and its environment. The MDP is defined by five components:

$$(S, A, P_a, R_a, \gamma), \tag{6}$$

where S represents the state space, A denotes the action space, P_a is the transition function, R_a is the reward function, and γ is the discount factor.

S is the state space, the coordinates of the whole body position of the soft robot are available in our proposed neural network model, so the contents of the state space can be specifically adjusted according to the task requirements. For the tracking task, where each state s consists of: the robot position coordinates $Y_t^{body} \in \mathbb{R}^{3n}$ (n is the number of piecewise key feature points on the whole body of the soft robot), the relative position vector from the current position to target position [25], $\Delta Y_t = Y^{tar} - Y_t^{body} \in \mathbb{R}^{3n}$ (Y^{tar} denotes the target whole-body key feature point coordinates), air pressure state for the three chambers (with values and directions) $P_t \in \mathbb{R}^6$, and time t. The state is hence defined as:

$$s_t = [Y_t^{body}, \Delta Y_t, P_t, t]. \tag{7}$$

The information contained in the state is fully observed by the agent. The target position is defined by a known deterministic and time-dependent trajectory function. Including both the

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

target position and the time step t in the state ensures that the state space satisfies the Markov property.

A is the action space, and the action $a_t \in A$ is defined as the change in pressure value $\Delta p \in \mathbb{R}^3$ at each interaction timestep t, which is set to be a continuous value in the range of -3 kPa to 3 kPa.

$$a_t = \Delta p \in \mathbb{R}^3 \tag{8}$$

$P_a(s, s')$ is the state transition function, representing the probability that taking action a will cause the system to transition from state s to state s'. In this work, the transition dynamic is defined based on the HWB-NN model:

$$s_{t+1} = T_{nn}(s_t, a_t). \tag{9}$$

$R_a(s, s')$ is the reward function, which represents the real-time expected reward corresponding to the transfer from state S to state S', and is used to assess whether the agent's behavior in performing the task is optimized towards the goal. In our tracking task, the reward is defined as the exponential of the negative Euclidean distance between the current and target positions in 3D Cartesian coordinates:

$$r_t = \exp(-\|Y^{\text{tar}} - Y^{\text{body}}\|). \tag{10}$$

The exponential nature of the reward function exhibits high sensitivity to small tracking errors, thereby promoting precise convergence. Its range is inherently bounded within $(0, 1]$, which enhances numerical stability. This formulation is also smooth and continuously differentiable, providing stable gradients for policy optimization. As this study focuses on tracking accuracy, actuation cost [22] is not considered. Such terms can be added for task-specific extensions.

B. Policy Training with Proximal Policy Optimization

PPO is adopted in this study as the reinforcement learning algorithm, as it has demonstrated strong empirical performance in robotic control. [26], [27], [28]. Additionally, in our work, we evaluated Soft Actor-Critic (SAC) [29], Trust Region Policy Optimization (TRPO) [30], and PPO on our soft robot control task. Compared to the other methods, PPO avoids second-order derivatives while maintaining stable updates through a clipped surrogate objective. In our experiments under identical conditions, PPO exhibited faster convergence and more stable learning behavior than TRPO and SAC. We also compute the ratio between the final reward and the task-specific theoretical maximum reward, with SAC, TRPO, and PPO achieving 73.0 %, 76.1 %, and 87.9 %, respectively. Therefore, we adopted PPO as our final control policy training algorithm.

PPO optimizes a clipped surrogate objective function, designed to limit the deviation between new and old policies. The objective function is defined as follows:

$$\begin{aligned} &L^{\text{CLIP}}(\theta) \\ &= \mathbb{E}_t \left[\min \left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t \right) \right], \end{aligned} \tag{11}$$

where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$ is the probability ratio, $\hat{A}_t$ is the advantage estimate at time t, and ϵ is a hyperparameter that controls the trade-off between policy flexibility and update stability.

The policy adopts an MLP architecture, with details provided in Table II. The actor and critic share the feature extraction layers, followed by separate output heads for action distribution and state value estimation.

The reinforcement learning environment is constructed based on the proposed HWB-NN model. At each time step, the environment receives an action input Δp, representing the pressure change in each pneumatic channel. The direction of pressure variation is detected from the sign of Δp, which is essential for capturing the pressure-dependent hysteresis behavior. Based on the magnitude and direction of the pressure in the current step, the HWB-NN model accurately predicts the whole-body position of the soft robot. A reward is then computed according to the predicted position and task objective. The reinforcement learning agent generates actions based on current observation to optimize task performance. All tasks are defined to lie entirely within the reachable workspace specified by the HWB-NN model.

C. Implementation Details

a) Action PID controller: The action output from the policy network is sent to a PID controller, which regulates the internal pressure of the soft robot to match the target pressure specified by the action. Considering the high inertia of the soft robotic system, the PID parameters are carefully tuned. The proportional gain $K_p = 130$, integral gain $K_i = 5$, and derivative gain $K_d = 30$ are selected to balance response speed and control stability. The control loop operates at a frequency of 100 Hz to ensure responsive and smooth actuation.

b) Parallel Environments: To accelerate training efficiency, multiple environments are executed in parallel. Increasing the number of environments improves policy update speed and sample diversity, while the final policy performance tends to converge to a similar level once a sufficient number of environments is reached. In our experiments, we used an NVIDIA RTX 4090 GPU with 64 parallel environments, which is the maximum number permitted by the available GPU memory. The average training time for the trajectory tracking task was 2.2 hours.

c) Hyperparameters: The configuration of hyperparameters used in training is listed in Table I.

TABLE I: PPO Hyperparameters

Hyperparameter	**Value**	**Hyperparameter**	**Value**
Total Timesteps	20M	Discount Factor, γ	0.99
Steps per Env	64	GAE, λ	0.95
Batch Size	4096	Entropy Coeff.	0.01
Mini-batches	1	Clip Range	0.2
Epochs	1	Policy Network	MLP, 2×64
Learning Rate	3e-4	Activation Function	Tanh
Annealing Strategy	Linear	Grad Clip Norm	None

V. EXPERIMENTS

To evaluate the advantages of the proposed HWB-NN model, we conduct accuracy comparisons on a test dataset.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

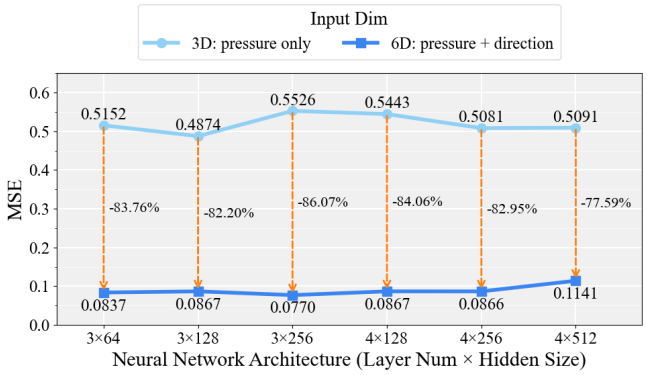


Fig. 4: Comparison of dynamic modeling performance across six neural network architectures under two input settings: 3D pressure-only and 6D pressure-plus-direction (proposed). The average MSE and corresponding percentage improvements are presented as evaluation metrics.

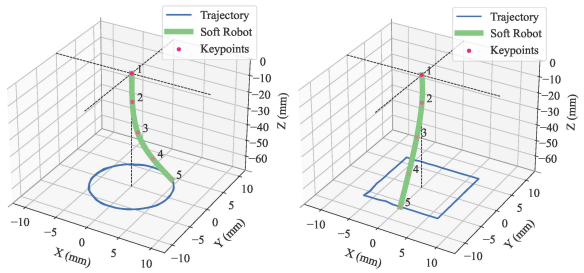
Furthermore, to assess the effectiveness of the proposed reinforcement learning framework in real-world control, the learned policy is deployed on the real soft robot and evaluated through a trajectory tracking task. To demonstrate the effectiveness of the system in complex tasks, we test it in a representative endoscopic surgery scenario, a typical application for soft robots.

A. Modeling Results and Analysis

To validate the effectiveness of the proposed HWB-NN model, two types of network are tested on the same test dataset, consisting of 1,000 points distributed throughout the action space. One type takes the proposed 6-dimensional input that includes both pressure values and their directional changes, while the other uses only 3 dimension pressure magnitudes, which is a conventional approach [12]. The resulting neural network dynamic models are evaluated on the test set using Mean Squared Error (MSE) to assess prediction accuracy. Each network is trained and tested five times with different random seeds. The average MSE values obtained across different network architectures are shown in Fig. 4. T-tests are conducted between the two input settings for each network. The resulting p-values for all architectures are below 0.01, indicating that the proposed method achieves significantly better accuracy than the conventional approach. The model architecture yielding the lowest MSE is selected, with an average training time of 651 seconds.

B. Simulation Results

The trained reinforcement learning policy is evaluated in simulation by performing a trajectory tracking task, where the soft robot is required to track a target trajectory with its end effector. The tracking performance is shown in Fig. 5 . The blue line indicates the motion path of end effector in simulation, while the green curve shows the predicted whole-body configuration of the robot. Five key feature points along the body are highlighted in magenta, demonstrating the availability of full-body pose information during tracking.



(a) Circle trajectory (b) Square trajectory

Fig. 5: Trajectory tracking performance of the soft robot in simulation.

C. Real-World Soft Robot Deployment Results

To evaluate the learned control policy, we deploy it on the real soft robot. The motion capture system is only used to record ground-truth positions for validation. We test four types of trajectories, including three planar trajectories (circle, square, and star) and one three-dimensional trajectory, the Viviani curve. We evaluate the tracking performance by computing the mean Euclidean distance (MED) between the actual positions of the robot's end effector and the target positions. Each trajectory experiment is repeated ten times, and the mean and standard deviation are calculated accordingly. In circular trajectory tracking experiments, we evaluate the effect of trajectory resolution on control performance by testing three angular resolutions: 0.2 rad, 0.13 rad, and 0.1 rad. The results indicate that higher resolution improves tracking accuracy but increases execution time due to more control steps. The trajectories followed by the soft robot's end effector in the real world and in simulation are compared in Fig. 6. The planar trajectories are projected onto the X–Y plane for visualization, while the spatial trajectory is displayed in a three-dimensional plot. Table II summarizes the target point intervals (resolutions), the MED errors in both simulation and real-world experiments, and the execution times during real-world operation.

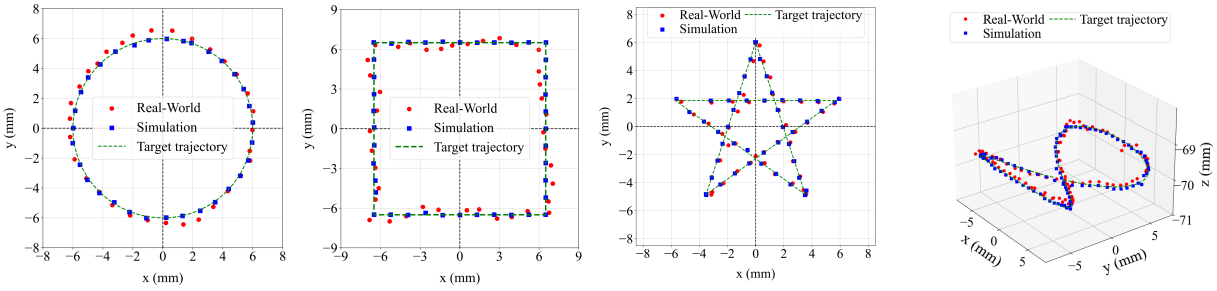
TABLE II: A Summary of the Target Point Resolutions, MED in Simulation and Real-World Experiments, and Real-World Execution Times Across Different Trajectories

Trajectory (Interval)	Simulation (mm)	Real-World (mm)	Time (s)
Circle (0.2 rad)	0.031 ± 0.001	0.211 ± 0.019	68
Circle (0.13 rad)	0.026 ± 0.002	0.183 ± 0.018	84
Circle (0.1 rad)	0.019 ± 0.001	0.147 ± 0.013	108
Square (1.3 mm)	0.019 ± 0.001	0.175 ± 0.021	75
Star (1.1 mm)	0.044 ± 0.001	0.240 ± 0.029	83
Viviani (0.13 rad)	0.092 ± 0.007	0.307 ± 0.046	134

D. Complex Task Whole-Body Control Experiments

To further demonstrate the advantages of our whole-body information-based control approach in complex tasks, we design a representative application task for laparoscopic surgery. In this scenario, a laser is mounted in the center of the soft robot. The objective is to control the robot so that the

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.



(a) Circle trajectory (b) Square trajectory (c) Star trajectory (d) Viviani trajectory

Fig. 6: Tracking results of the soft robot end effector on (a) Circular, (b) Square, (c) Star, (d) Viviani trajectories in both real-world and simulation experiments.

laser beam intersects a distant plane and follows a predefined trajectory to ablate pathological tissues.

This task requires accurate estimation of the laser pointing direction, which depends on the soft robot's global curvature. Conventional modeling and control methods are insufficient to handle this due to their limitation in capturing whole-body deformation. Using our hysteresis-aware whole-body modeling approach, the simulation environment can accurately predict the laser projection point on the remote plane. Combined with the PPO algorithm, which is suited for high-dimensional state spaces, the framework can effectively learn control policies to accomplish this complex task. The average Euclidean distance (L2 norm) between the intersection trajectory and the target trajectory in the simulation experiments is 0.0419 mm, as shown in Fig. 7 (a).

To evaluate the performance of deploying the proposed soft robot system in the abdominal cavity, we perform a simulated liver laser ablation experiment using a mannequin model. Considering the limited visibility commonly encountered in laparoscopic surgeries, the system is designed with two soft robots: one equipped with a laser module and the other with an endoscope, as shown in Fig. 7 (c).

The endoscope-equipped soft robot first bends toward the target to provide a view of the surgical site. Subsequently, the laser-equipped soft robot performs circular ablation around the lesion, as shown in Fig. 7 (d). The laser-equipped soft robot accurately completed a full circular ablation around a 20 mm-diameter lesion within 60 seconds. The experimental results demonstrate that the proposed control method achieves high accuracy, stability, and coordination in real-world application scenarios.

VI. Discussion and Conclusion

Summary and Contributions: In this paper, we propose a reinforcement learning control approach for soft robots based on a Hysteresis-aware Whole-Body Neural Network(HWB-NN) model. To address tasks with morphological constraints, a hysteresis-aware piecewise key feature point neural network model is proposed, which represents the whole-body state of the soft robot by modeling a limited set of key feature points. To account for the hysteresis characteristics of soft robots, the proposed neural network model integrates pressure direction

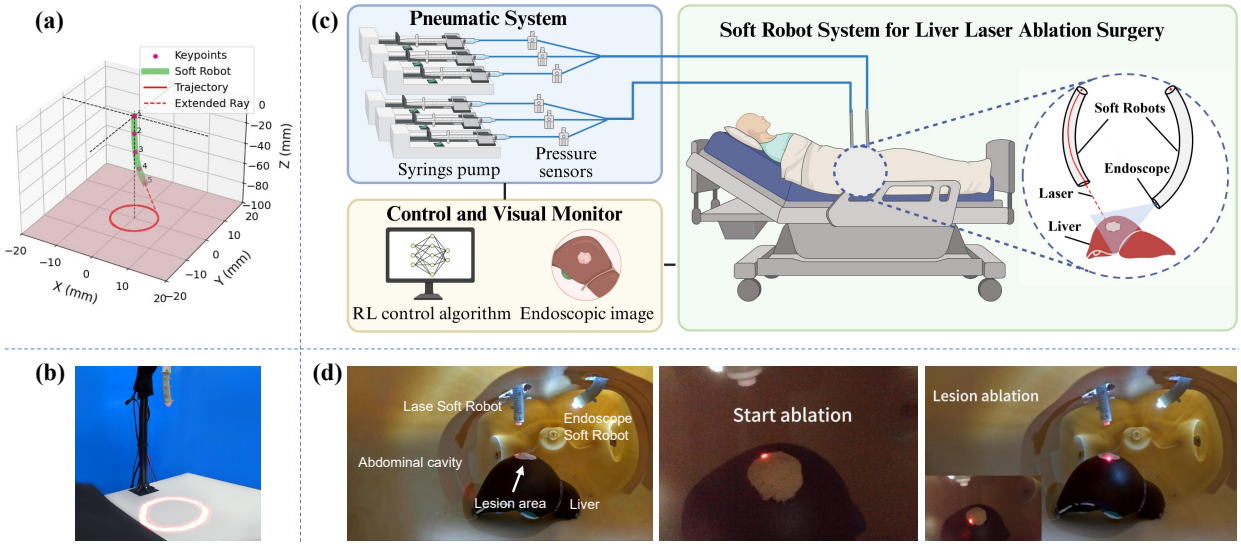


Fig. 7: (a) Laser pointing experiment in simulation. (b) Laser pointing experiment in real-world. (c) Overview of the two-soft-robot system for laparoscopic surgery. (d) Time-sequenced snapshots of the liver laser ablation procedure.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

information, which effectively reducing the prediction errors associated with historical pressure paths. To achieve whole-body control with high-dimensional state space, we construct a parallelized simulation environment based on the accurate HWB-NN model and apply the reinforcement learning algorithm to efficiently train the control policy.

Limitations and Future Work: The current modeling and control framework is developed under quasi-static assumptions and may not generalize well to high-speed motion scenarios. Future work may investigate temporal modeling approaches to strengthen dynamic modeling capabilities. The proposed data-driven modeling reduces the simulation-to-reality gap and generalizes to soft robots with various materials and structures, but requires large training datasets. Future work will improve data efficiency and streamline data collection. The current method is limited to single-agent settings and has not yet explored force-aware manipulation. Future work may incorporate force sensing and multi-agent reinforcement learning frameworks.

ACKNOWLEDGMENT

Fig. 7(c) was created with BioRender.com and is used with permission under a BioRender Publication License.

REFERENCES

[1] F. Pique, H. T. Kalidindi, L. Fruzzetti, C. Laschi, A. Menciassi, and E. Falotico, "Controlling Soft Robotic Arms Using Continual Learning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 5469–5476, Apr. 2022.

[2] T. Čakurda, M. Trojanová, P. Pomin, and A. Hošovský, "Deep Learning Methods in Soft Robotics: Architectures and Applications," *Adv. Intell. Syst.*, p. 2400576, Nov. 2024.

[3] S. Terryn, J. Brancart, D. Lefeber, G. Van Assche, and B. Vanderborght, "Self-healing soft pneumatic robots," *Sci. Robot.*, vol. 2, no. 9, p. eaan4268, 2017.

[4] Z. Jiao, C. Ji, J. Zou, H. Yang, and M. Pan, "Vacuum-powered soft pneumatic twisting actuators to empower new capabilities for soft robots," *Adv. Mater. Technol.*, vol. 4, no. 1, p. 1800429, 2019.

[5] D. Wu, X. T. Ha, Y. Zhang, M. Ourak, G. Borghesan, K. Niu, F. Trauzettel, J. Dankelman, A. Menciassi, and E. V. Poorten, "Deep-Learning-Based Compliant Motion Control of a Pneumatically-Driven Robotic Catheter," *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 8853–8860, Oct. 2022.

[6] B. Zhang, P. Yang, X. Gu, and H. Liao, "Laser Endoscopic Manipulator Using Spring-Reinforced Multi-DoF Soft Actuator," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7736–7743, Oct. 2021.

[7] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1661–1683, 2010.

[8] Y. Zhou, L. M. Headings, and M. J. Dapino, "Modeling of fluidic pre-stressed composite actuators with application to soft robotic grippers," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2166–2178, 2022.

[9] B. A. Jones and I. D. Walker, "Practical kinematics for real-time implementation of continuum robots," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1087–1099, 2006.

[10] M. S. Nazeer, C. Laschi, and E. Falotico, "RL-Based Adaptive Controller for High Precision Reaching in a Soft Robot Arm," *IEEE Trans. Robot.*, vol. 40, pp. 2498–2512, 2024.

[11] H. Zheng, S. Pinzello, B. G. Cangan, T. J. K. Buchner, and R. K. Katzschmann, "Vision-Based Online Key Point Estimation of Deformable Robots," *Adv. Intell. Syst.*, vol. 6, no. 10, p. 2400105, Oct. 2024.

[12] G. Lou, C. Wang, Z. Xu, J. Liang, and Y. Zhou, "Controlling Soft Robotic Arms Using Hybrid Modelling and Reinforcement Learning," *IEEE Robot. Autom. Lett.*, vol. 9, no. 8, pp. 7070–7077, Aug. 2024.

[13] Y.-Y. Wu and N. Tan, "Model-less feedback control for soft manipulators with jacobian adaptation," in *Proc. Int. Symp. Autom. Syst.* IEEE, 2020, pp. 217–222.

[14] Y. Zhang, J. Gao, H. Yang, and L. Hao, "A novel hysteresis modelling method with improved generalization capability for pneumatic artificial muscles," *Smart Mater. Struct.*, vol. 28, no. 10, p. 105014, Oct. 2019.

[15] Z. Chen, F. Renda, A. Le Gall, L. Mocellin, M. Bernabei, T. Dangel, G. Ciuti, M. Cianchetti, and C. Stefanini, "Data-Driven Methods Applied to Soft Robot Modeling and Control: A Review," *IEEE Trans. Autom. Sci. Eng.*, vol. 22, pp. 2241–2256, 2025.

[16] A. Centurelli, L. Arleo, A. Rizzo, S. Tolu, C. Laschi, and E. Falotico, "Closed-Loop Dynamic Control of a Soft Manipulator Using Deep Reinforcement Learning," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4741–4748, Apr. 2022.

[17] K. Zhou, B. Mao, Y. Zhang, Y. Chen, Y. Xiang, Z. Yu, H. Hao, W. Tang, Y. Li, H. Liu, X. Wang, X. Wang, and J. Qu, "A Cable-Actuated Soft Manipulator for Dexterous Grasping Based on Deep Reinforcement Learning," *Adv. Intell. Syst.*, vol. 6, no. 10, p. 2400112, Oct. 2024.

[18] M. A. Graule, T. P. McCarthy, C. B. Teeple, J. Werfel, and R. J. Wood, "SoMoGym: A Toolkit for Developing and Evaluating Controllers and Reinforcement Learning Algorithms for Soft Robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4071–4078, Apr. 2022.

[19] S. Satheeshbabu, N. K. Uppalapati, G. Chowdhary, and G. Krishnan, "Open loop position control of soft continuum arm using deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom.* IEEE, 2019, pp. 5133–5139.

[20] T. G. Thuruthel, E. Falotico, F. Renda, and C. Laschi, "Model-Based Reinforcement Learning for Closed-Loop Dynamic Control of Soft Robotic Manipulators," *IEEE Trans. Robot.*, vol. 35, no. 1, pp. 124–134, Feb. 2019.

[21] F. Connolly, P. Polygerinos, C. J. Walsh, and K. Bertoldi, "Mechanical Programming of Soft Actuators by Varying Fiber Angle," *Soft Robot.*, vol. 2, no. 1, pp. 26–32, Mar. 2015.

[22] Y. Huang, A. Y. Alkayas, J. Shi, F. Renda, H. Wurdemann, and T. G. Thuruthel, "Predicting Interaction Shape of Soft Continuum Robots using Deep Visual Models," in *2024 Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.* Abu Dhabi, United Arab Emirates: IEEE, Oct. 2024, pp. 11 381–11 387.

[23] M. L. Puterman, "Markov decision processes," *Handb. Oper. Res. Manag. Sci.*, vol. 2, pp. 331–434, 1990.

[24] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[25] J. Chen, C. Yu, Y. Xie, F. Gao, Y. Chen, S. Yu, W. Tang, S. Ji, M. Mu, Y. Wu, H. Yang, and Y. Wang, "What matters in learning a zero-shot sim-to-real rl policy for quadrotor control? a comprehensive study," *IEEE Robot. Autom. Lett.*, vol. 10, no. 7, pp. 7134–7141, 2025.

[26] U. Berdica, M. Jackson, N. E. Veronese, J. Foerster, and P. Maiolino, "Reinforcement Learning Controllers for Soft Robots Using Learned Environments," in *2024 Proc. IEEE Int. Conf. Soft Robot.* San Diego, CA, USA: IEEE, Apr. 2024, pp. 933–939.

[27] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Sci. Robot.*, vol. 4, no. 26, p. eaau5872, Jan. 2019.

[28] N. Naughton, J. Sun, A. Tekinalp, T. Parthasarathy, G. Chowdhary, and M. Gazzola, "Elastica: A Compliant Mechanics Environment for Soft Robotic Control," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3389–3396, Apr. 2021.

[29] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn.*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 1861–1870. [Online]. Available: https://proceedings.mlr.press/v80/haarnoja18b.html

[30] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 1889–1897. [Online]. Available: https://proceedings.mlr.press/v37/schulman15.html