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

Learning-Based Kinematic Modeling for Concentric Tube Robot: Addressing its Nonlinearity and Snapping Behavior

Gowoon Jeong, and Seong Young Ko, *Member, IEEE*

***Abstract*—The Concentric Tube Robot (CTR) has great promise for minimally invasive surgery. However, accurately modeling nonlinear and history-dependent behaviors remains a significant challenge. This letter proposes a learning-based forward and inverse kinematics model that accounts for the history dependence and nonlinearities of CTR, including the snapping behavior. A lightweight LSTM-MLP hybrid neural network with an input buffer and directional parameters was used to train forward and inverse kinematics models for 4-degree-of-freedom (DOF) CTR. The model was validated by comparing its predictions with actual values and results from a conventional torsional-compliant model (TCM) across random points, rotational trajectories, and arbitrary paths. This validation successfully demonstrated the model's ability to capture snapping behavior. For forward kinematics, the model achieved a Root Mean Square Error (RMSE) of 0.69 mm and 0.16° with a computation time of 0.831±0.200 ms. The inverse kinematics model achieved an RMSE of 1.22 mm and 2.46° with a computation time of 0.816±0.200 ms. The proposed method improves the accuracy and speed of kinematic modeling by capturing nonlinear behaviors, such as snapping and hysteresis. The lightweight system ensures accurate real-time control and offers a safer and more reliable solution for microsurgical applications.**

***Index Terms*—Machine Learning for Robot Control, Kinematics, Surgical Robotics: Steerable Catheters/Needles.**

I. Introduction

THE concentric tube robot (CTR) comprises pre-curved, super-elastic tubes nested within one another [1]. The shape of the tube depends on relative rotation and translation, making it suitable for minimally invasive surgery [2]. However, the nonlinear interaction between these super-elastic tubes complicates modeling, particularly the mathematical representation of physical factors including gap and friction between tubes, which poses significant challenges for real-time microsurgery applications. The conventional approach to modeling CTR is the torsional-compliant model (TCM), which addresses the bending and torsional effects [1]. Moreover, although bounding kinematic solutions using coulomb friction models have been proposed, these models ignore key nonlinear factors such as cross-sectional shear stress, axial elongation, and nonlinear material properties [3]. In inverse kinematics models, analytical solutions are unavailable, computation burdens are high, and solution convergence is not guaranteed, making them unsuitable for real-time control [4,5]. Another critical problem arises when the robot reaches the bifurcation point, where the accumulated twisting energy is released abruptly, causing the tubes to snap to a distant position. Several methods have been proposed to mitigate snapping through local bifurcation approaches [6], energy reduction patterns [7, 8], and design techniques [6, 9]. However, the use of CTR with snapping behavior has received relatively limited attention. Accurate kinematic analysis of mechanisms with snapping risks can expand applications, such as improving highly precise minimally invasive surgery in constrained spaces by predicting, avoiding, or inducing snapping motions for path planning. Finally, the kinematics of CTR are history-dependent because of hysteresis and clearance [1, 10]. Hysteresis errors vary with direction, and kinematics change according to the previous path. Therefore, achieving accurate CTR kinematics considering nonlinearity, snapping behavior, and history-dependent characteristics can advance microsurgery and real-time control across more diverse applications.

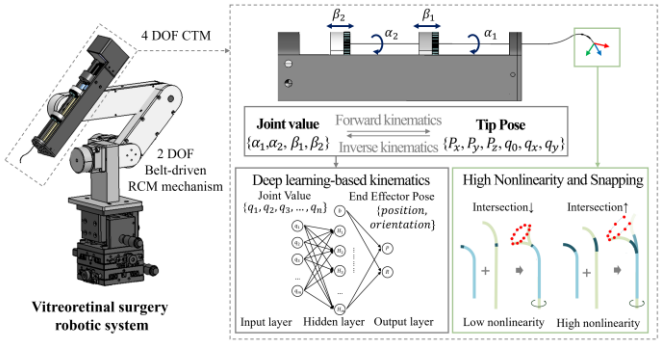

Fig. 1. Concept of learning-based kinematics for CTR.

Methods that employ closed-loop controllers have been

Manuscript received: December 31, 2024; Revised March 28, 2025; Accepted May 2, 2025.

This paper was recommended for publication by Editor Pietro Valdastri upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Technology Innovation Program (20023168, Development of clinician collaborative robot platform technology for 3 types of pain interventional procedures based on crossdrug handler and instrument modules) funded By the Ministry of Trade, Industry & Energy (MOTIE, Korea). (Corresponding author: Seong Young Ko.)

G. Jeong is with the Graduate School Department of Mechanical Engineering, Chonnam National University, Gwangju, 61186, Republic of Korea (e-mail: gjeong@jnu.ac.kr).

S. Y. Ko is with the School of Mechanical Engineering, Chonnam National University, Gwangju, 61186, Republic of Korea (e-mail: sko@jnu.ac.kr).

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.

proposed to prevent snapping by considering force/velocity manipulability [11] or by employing a model predictive controller [12]. However, these approaches are typically challenging to implement without sensors for feedback. Deep learning has recently emerged as a promising approach for addressing the nonlinearity of continuum robots and has demonstrated feasibility in simulation [13-15] and experiments with real robots [16]. Learning-based kinematic approaches enable the prediction of the CTR shape [17] and the solutions of forward and inverse kinematics [18, 19]. Despite these advances, these models are often trained on random points without considering the robot's history or extreme nonlinear states. Learning-based methods offer significant advantages by capturing nonlinearity and history-dependent time-series characteristics. The kinematics of CTRs, including the snapping and hysteresis effects, have been validated through reduced-order dynamic simulations [20]. In addition, deep learning has proven effective in capturing the hysteresis characteristics of tendon-driven mechanisms [21-23] and soft robots [24].

This letter proposes a lightweight deep learning-based forward and inverse kinematic model that accurately predicts nonlinearity, hysteresis, and snapping. A lightweight Long Short-Term Memory – Multilayer Perceptron (LSTM-MLP) hybrid model was designed to handle time-series data and nonlinearity using a history input buffer with a relative rotation indicator. To our knowledge, this is the first learning-based kinematic model that considers the robot's history-dependent state while predicting snapping occurrence and severity while accounting for hysteresis. The proposed approach enhances the safe operation of CTRs in microsurgery applications, offering high precision and applicability in various medical scenarios. Additionally, the lightweight nature of the model makes it suitable for real-time control.

II. Nonlinearity of the Concentric Tube Robot

A. Concentric Tube Robot for Eye Surgery

In our previous study, we developed a robotic system for vitreoretinal surgery incorporating a 4-DOF CTR and a 2-DOF belt-driven remote center of motion (RCM) mechanism [25], as shown in Fig. 1. This mechanism maximizes the robot tip's reachable workspace within the eye while minimizing link motions outside the eyeball. The CTR comprised two pre-curved nitinol tubes and was designed considering the eyeball's 24 mm diameter and anatomical structure, as shown in Table I. The continuum robot's curvature and shape vary based on the rotation of the inner tube and translation of the inner tube α_1, β_1, and the rotation and translation of the outer tube α_2, β_2. As the curvature or intersection section increases, the robot's nonlinearity increases. The contributing factors include manufacturing defects, significant friction between the tubes, and hardware tolerances. Perfect hardware is the ideal solution; however, achieving such precision is challenging. Although the robot is compact and designed with high curvature to suit retinal surgery, snapping—though rare—can still occur within its workspace. Retinal surgery requires high precision within a range of a few dozen micrometers, making unpredictable movements, such as snapping, a significant safety concern. Thus, predicting snapping events while considering the robot's history and maximizing its usable range of motion is critical for ensuring safe operation.

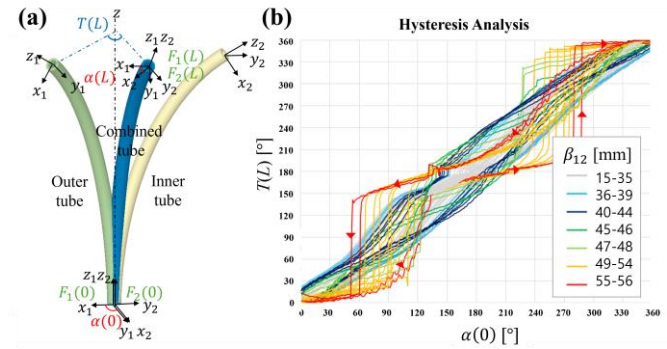

Fig. 2. Nonlinearity analysis: (a) coordinate system of the CTR and (b) hysteresis analysis result.

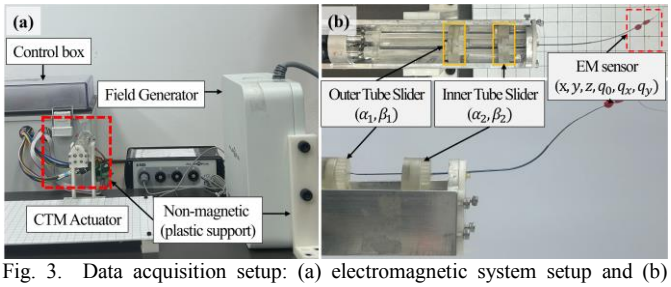

Fig. 3. Data acquisition setup: (a) electromagnetic system setup and (b) 4-DOF CTR setup.

TABLE I
DETAILS OF EACH CONCENTRIC TUBE

	Outer tube		*Inner tube*	
	Symbol	**Dimension**	**Symbol**	***Dimension***
Outer diameter	OD_1	0.635mm	OD_2	0.4 mm
Inner diameter	ID_1	0.48mm	ID_2	0.2 mm
Radius of curvature	r_1	24 mm	r_2	27 mm
Curved part length	$l_{1,1}$	14.5 mm	$l_{2,1}$	15 mm
Straight part length	$l_{1,2}$	37 mm	$l_{2,2}$	57 mm

B. Nonlinearity Analysis

To analyze the nonlinear motion of the CTR, the coordinate frames of the outer, inner, and combined tubes were defined, as shown in Fig. 2(a). Based on the TCM proposed in [1], the relative z-axis twist angle α(s) between these tube coordinate frames $F_i(s)$ varied from a maximum α(0) at the base to a minimum α(L) at the tip. To simplify the kinematics modeling based on the deep learning model intuitively, we defined $T(L)$ as the angle between the outer tube base and the robot tip relative to the z-axis. This aligns with the snapping behavior caused by nonlinearity and solution multiplicity discussed in [1]. Fig. 3 shows the data acquisition setup used to capture the robot's joint values and tip pose. Four DC motors (Maxon Motor Co. RE8) were used to drive the tube's translation via a lead screw and rotation through gear transmission [25]. Joint values were measured using MR encoders with a position controller (EPOS4 Compact 24/1.5 CAN). To measure the position of the tip, an electromagnetic sensor (NDI, Aurora 5-DOF sensor) is attached to the tip. To minimize noise and reduce gravity effects, a plastic fixture and a thin tube were

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

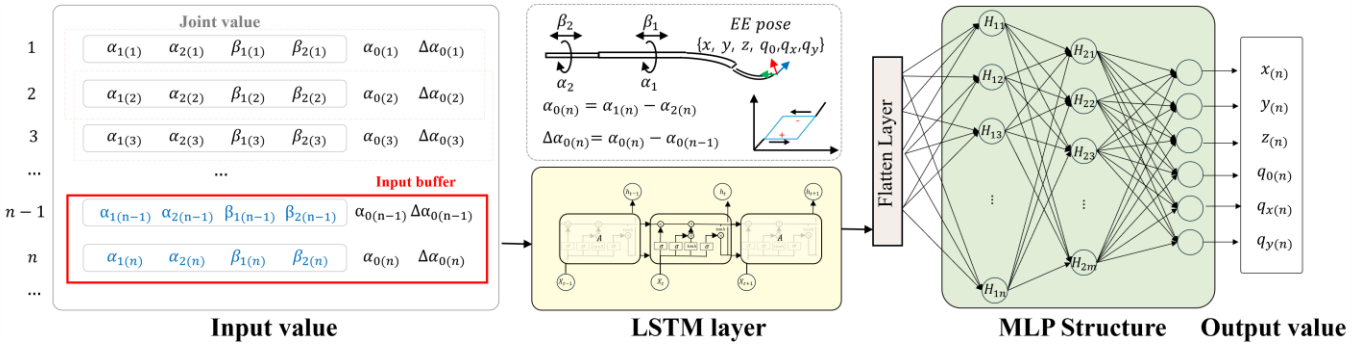

Fig. 4. Architecture of the lightweight hybrid LSTM-MLP kinematics model for CTR.

used to secure the sensor. Each joint value was obtained by averaging five consecutive readings from the sensor to minimize measurement errors. To analyze tube nonlinearity, the outer tube was fixed while the inner tube was rotated bi-directionally in 1° increments over 360° with varying relative translations. This setup is also used for data collection during model training in Section IV, which induces outer tube rotation.

As α(0) rotated from 0° to 360° and 360° to 0°, the relationship between $T(L)$ was analyzed, as shown in Fig. 2(b). $\beta_{12} = \beta_2 - \beta_1$ denotes the relative translation of the inner tube, measured when the outer tube is fixed. For small β_{12} values, the curved section of the outer tube intersects with the straight or less curved section of the inner tube, resulting in a small hysteresis error, as shown in the gray or blue lines in Fig. 2(b). As β_{12} increases, the overlap of the more curved sections intensifies the nonlinearity, resulting in higher hysteresis errors and snapping phenomena. In these cases, the accumulated energy is released abruptly, causing sudden displacements. In all scenarios, the hysteresis is evident through path differences depending on the direction of rotation. In contrast to mathematical modeling approaches that define snapping by $a(L)$, as in [1], the proposed approach relies on deep learning model to identify snapping events that are instances when $T(L)$ changes by more than 1 0 degrees. Although previous studies in literature have primarily addressed smooth kinematic regions, the proposed learning-based kinematic model captures both smooth transitions and snapping phenomena, as outlined in the following Section.

III. Method

A. Data Acquisition

Data for learning-based kinematics was collected using the experimental setup described in Section II.B. The data were collected by varying the relative translation and rotating the outer tube in 10-degree increments, while the inner tube performed a full rotation in both directions, thereby capturing snapping scenarios. This approach contrasts with that of [26], which used approximately 100,000 random data points to train the kinematics of CTR in SE(3), without considering snapping scenarios. The data acquisition process required approximately 21 hours, yielding a total dataset of 212,527 samples. Outliers caused by encoder errors or robot positions outside the workspace were removed. The joint values are represented as $\{\alpha_1, \alpha_2, \beta_1, \beta_2\}$, with units in degrees and millimeters, while the tip pose is denoted as $\{x, y, z, q_0, q_x, q_y\}$, where the position is measured in millimeters in Cartesian coordinates, and orientation is expressed as a quaternion. The 5-DOF sensor was used, the yaw component was ignored since q_z was always 0. The approximation error defined in [26] was used as the translation and rotation errors of the forward kinematics and inverse kinematics. The Cartesian space error is expressed as the distance difference, considering the antipodal property of quaternions, and is described in terms of the Euclidean distance in the joint coordinate system.

B. Model Design

The requirements for the learning-based kinematics model were defined to ensure optimal performance. For surgical applications, the model must achieve an error accuracy of less than 1 mm under both high and low nonlinearity conditions. Additionally, for real-time control, the model must be lightweight and capable of handling time-series data to consider history dependence.

We propose a lightweight LSTM-MLP hybrid model for learning both the kinematics of the CTR, as shown in Fig. 4. For forward kinematics, the input value is the joint value, and the output is the tip pose. To enhance model performance, the additional parameter $\alpha_0(n)$ was incorporated in the input layer for the n^{th} joint input as follows:

$$\alpha_0(n) = \alpha_1(n) - \alpha_2(n) \quad (1)$$

$$\Delta\alpha_0(n) = \alpha_0(n) - \alpha_0(n-1) \quad (2)$$

Where $\alpha_0(n)$and $\Delta\alpha_0(n)$ denote the relative rotation and its derivative of the robot, respectively. This parameter is a key feature to capture directional characteristics and intricate nonlinearities. The combination of $\Delta\alpha_0(n)$ and the rectified linear unit (ReLU) activation function significantly affected the model's learning process. To incorporate history dependency, an input history buffer is employed to include both current and previous values. These inputs are processed through a Long Short-Term Memory (LSTM) layer, which is adept at learning both history dependencies and hysteresis. The output from the LSTM layer is passed through a flattening layer to reshape it into a format compatible with the Multi-Layer Perceptron (MLP). The MLP maps the nonlinear kinematics to predict the robot's tip pose. Consequently, the overall model can be represented by (3)~(7).

$$h_{LSTM1} = LSTM1(X) \quad (3)$$

$$h_{LSTM2} = LSTM2(h_{LSTM1}) \quad (4)$$

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

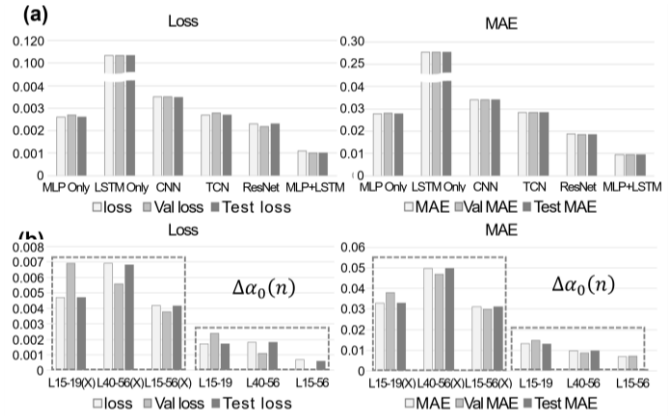

Fig. 5. Learning results: (a) model comparison, (b) effect of parameters on directional learning. The x-axis indicates relative translation (β_{12}).

$$h_{Flatten} = Flatten(h_{LSTM2}) \quad (5)$$
$$h_{Dense1} = ReLU(h_{Flatten}) \quad (6)$$
$$y_{pred} = ReLU(h_{Dense1}) \quad (7)$$

X represents the input, y_{pred} denotes the output, and h represents the hidden layers that consist of the model.

C. Model Optimization

This section details the design considerations and optimization process of the proposed model. The performance of the proposed LSTM-MLP model was compared with that of the existing models to evaluate its structure and performance using the robot data discussed in Section III.A. The comparisons were based on Mean Absolute Error (MAE), with the training environment outlined in Section IV.A. The following six models were considered: MLP, LSTM, Convolutional Neural Networks (CNN), Temporal Convolutional Networks (TCN), Residual Network (ResNet), and LSTM-MLP. The feedforward MLP is widely employed in the kinematic modeling of CTR due to its effectiveness in capturing nonlinearity [18]. When combined with an input buffer, MLPs can account for historical dependencies. Recurrent Neural Networks (RNNs), which excel at processing time-series data, have demonstrated effectiveness in learning hysteresis in tendon-driven mechanisms [21]. Long Short-Term Memory (LSTM), a specialized type of RNN, has been demonstrated to successfully compensate for hysteresis in tendon-sheath mechanisms when combined with the Preisach model [27]. Inspired by studies that obtained kinematics from concentric tube shapes, additional models—CNN, TCN, and ResNet—were included for comparison [17, 19].

The training results shown in Fig. 5(a) indicate that the LSTM-MLP hybrid model achieved the highest performance and exhibited the lowest error among the evaluated models. Although ResNet exhibited low error, its model size was relatively large. LSTM was expected to effectively capture hysteresis characteristics; however, it struggled to adequately account for the dependency between the forward and inverse kinematics hysteresis. CNN and TCN were excluded from further consideration due to their heavier architecture and comparatively higher loss and MAE. Consequently, the LSTM-MLP hybrid model was selected.

The effect of $\Delta\alpha_0(n)$ was evaluated by comparing the training results with and without this directional parameter. Fig. 6 shows the results with the actual and predicted positions marked in red and blue, respectively (both position and orientation are shown). Three scenarios were tested: $\beta_{12} = 15$, representing relatively low nonlinearity, $\beta_{12} = 35$, with higher

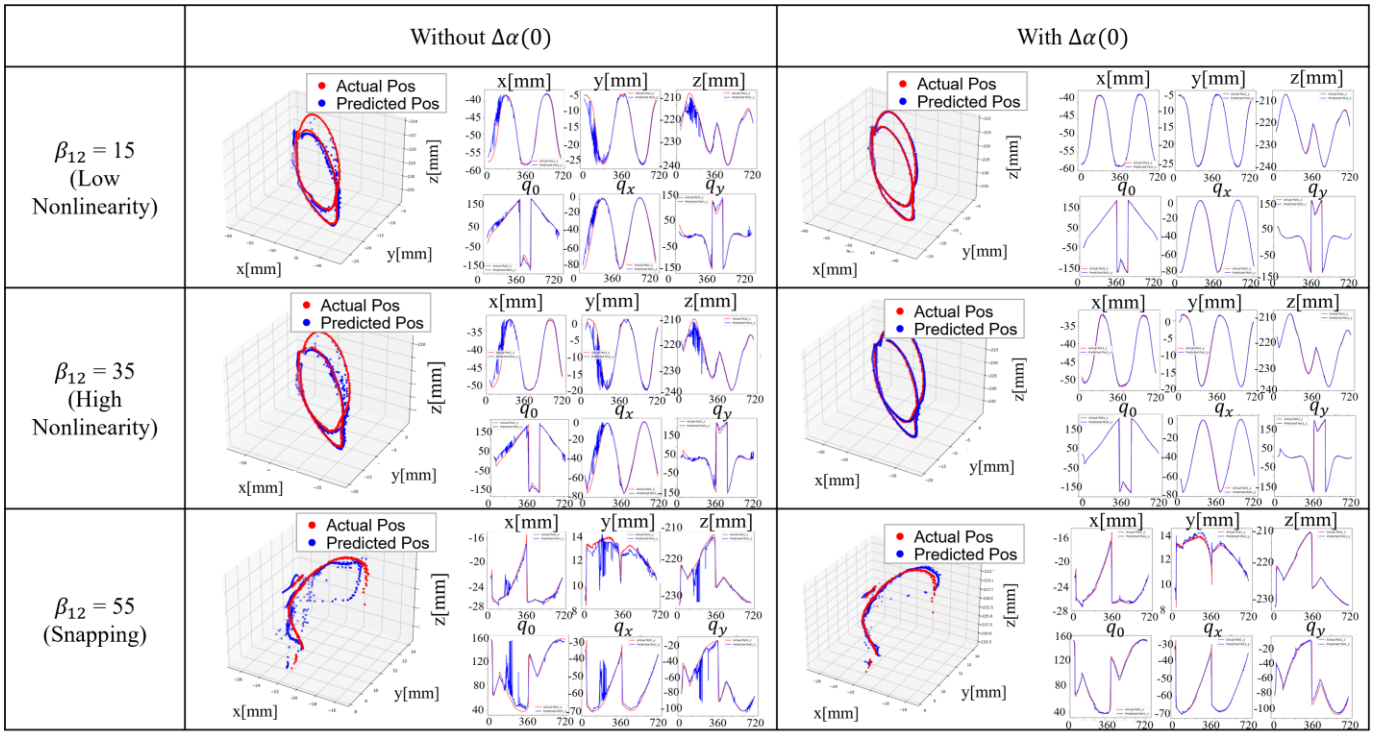

Fig. 6. Learning results demonstrating the impact of the proposed parameters on accounting for history-dependent effects. β_{12} = 15 implies low nonlinearity, β_{12} = 35 implies high nonlinearity, and β_{12} = 55 implies snapping situation. The actual tip position measured by the electromagnetic sensor (red) and the predicted position (blue) are shown.

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

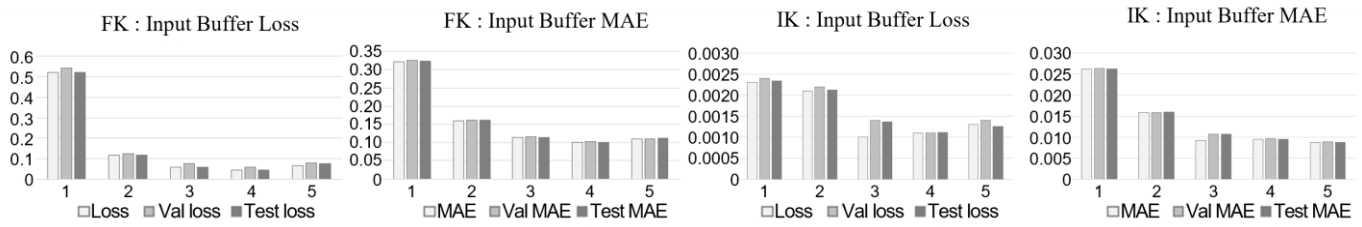

Fig. 7. Model optimization and forward kinematics (FK) and inverse kinematics (IK) training results. The x-axis represents the number of input buffers (IB).

TABLE II
TRAINING RESULTS IN FK AND IK BASED ON THE NUMBER OF INPUT BUFFERS

	Forward kinematics					Inverse kinematics				
	IB=1	**IB= 2**	**IB= 3**	**IB= 4**	**IB= 5**	**IB= 1**	**IB= 2**	**IB= 3**	**IB= 4**	**IB= 5**
Training Loss	0.5214	0.1800	0.0586	0.0452	0.0648	0.0023	0.0021	0.0010	0.0011	0.0013
Training MAE	0.3224	0.1599	0.1129	0.0999	0.1096	0.0262	0.0159	0.0093	0.0095	0.0088
Val loss	0.5405	0.1250	0.0780	0.0591	0.0813	0.0024	0.0022	0.0014	0.0011	0.0014
Val MAE	0.3253	0.1622	0.1153	0.1021	0.1108	0.0263	0.0159	0.0108	0.0096	0.0089
Test loss	0.5215	1.1822	0.0601	0.0463	0.0777	0.0023	0.0021	0.0013	0.0011	0.00126
Test MAE	0.3243	0.1613	0.1135	0.1006	0.1115	0.0262	0.0159	0.0108	0.0095	0.0087

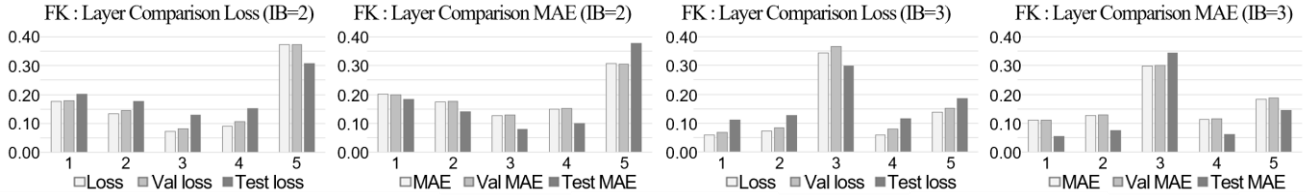

Fig. 8. Model optimization and the result in FK based on the various architectural combinations. The selected model corresponds to the cases when IB = 2 and IB = 3. The x-axis represents the number of input buffers (IB). S1 to S5 correspond to the selected structures as provided in Table III.

TABLE III
TRAINING RESULTS IN FK AND IK BASED ON THE VARIOUS ARCHITECTURAL COMBINATIONS

	Forward kinematics (IB = 2)					Forward kinematics (IB = 3)				
	S1	**S2**	**S3**	**S4**	**S5**	**S1**	**S2**	**S3**	**S4**	**S5**
Training loss	0.1759	0.1328	0.0716	0.0907	0.3738	0.0593	0.0725	0.3434	0.0588	0.1379
Training MAE	0.2009	0.1751	0.1277	0.1489	0.3073	0.1105	0.1269	0.2982	0.1135	0.1842
Val loss	0.1792	0.1437	0.0823	0.1063	0.3720	0.0680	0.0845	0.3654	0.0805	0.1521
Val MAE	0.1994	0.1773	0.1297	0.1506	0.3062	0.1121	0.1284	0.3015	0.1164	0.1866
Test loss	0.2006	0.1757	0.1289	0.1505	0.3070	0.1111	0.1276	0.2990	0.1152	0.1857
Test MAE	0.1842	0.1403	0.0795	0.0986	0.3781	0.0537	0.0753	0.3442	0.0611	0.1442
Position error[mm]	1.05	0.92	0.69	0.77	1.51	0.57	0.67	1.44	0.61	0.93
Orientation error [°]	0.58	0.20	0.16	0.17	0.40	0.13	0.14	0.53	0.14	0.19
Structure (S1~S5)	LSTM 128					LSTM 128				
	LSTM 64	LSTM 64	LSTM 128	ReLU 128	ReLU 64	LSTM 64	LSTM 64	LSTM 128	ReLU 128	ReLU 64
	Flatten	Flatten	Flatten	Flatten	Flatten	Flatten	Flatten	Flatten	Flatten	Flatten
	ReLU 32	ReLU 32	ReLU 64	LSTM 64	LSTM 32	ReLU 32	ReLU 32	ReLU 64	LSTM 64	LSTM 32
	ReLU 6	ReLU 6	ReLU 6	ReLU 6	ReLU 6	ReLU 6	ReLU 6	ReLU 6	ReLU 6	ReLU 6

nonlinearity, and $\beta_{12} = 55$, where snapping occurs during a full rotation in different rotation directions in each case. When the combination of $\Delta\alpha_0(n)$ and the ReLU activation function was used, the loss decreased by 72.24%, and the MAE decreased by 75.32%, as shown in Fig. 5(b). This approach accurately captures the differences based on direction.

The grid search method was used to optimize the proposed model's structure. The LSTM-first and MLP-first configurations were tested for the sequences of LSTM and MLP layers, and the number of layers was limited to two or fewer to ensure a lightweight model. The number of hidden nodes was set to 128, 64, and 32, and the number of input buffers ranged from 1 to 5. After training all combinations over 500 epochs, the best-performing model was selected. The data were split into training, validation, and test sets at a 70:15:15 ratio, with normalization applied in the range [0, 1]. Some combinations and training results are shown in Fig. 7. Fig. 7 and Table II compare the number of input buffers for forward and inverse kinematics, indicating that two or more input buffers are sufficient. The increased error with 4-5 input buffers may stem from the model's difficulty in handling long-term dependencies and outdated information, which can negatively affect the prediction of sudden movements. Models with various structures were trained for each input buffer, as shown in Fig. 8 and Table III, with the position and orientation reported in millimeters and degrees. Considering the compatibility and lightweight design for forward and inverse kinematics, the final model was determined to include two

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

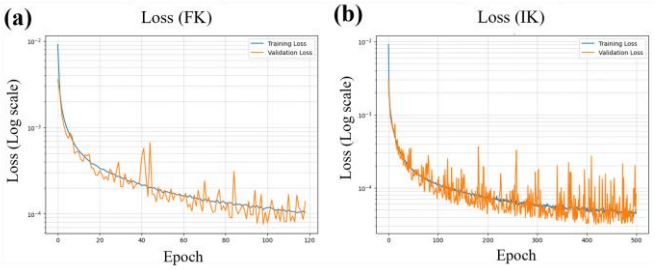

Fig. 9. Training and validation loss: (a) forward kinematics and (b) inverse kinematics. (Since the loss is similar to MAE, only the loss is shown.)

input buffers, an LSTM layer with 128 hidden nodes, and an MLP layer with 64 ReLU units. The results of the trained forward and inverse kinematics using the optimized model are shown in Fig. 9.

IV. Validation and Discussion

A. Forward Kinematics

The model was trained on an NVIDIA GeForce GTX 1050 Ti using TensorFlow 2.10 with Python 3.8. The training was performed using the Adam optimizer with a learning rate of 0.001 for 500 epochs and a batch size of 32, with early stopping applied to prevent overfitting. As shown in Fig. 9, the forward kinematics achieved a position error of 0.69 mm and an orientation error of 0.16°. The performance was evaluated using the MAE loss function. To validate the forward kinematics, three experiments were conducted: snapping and hysteresis verification, evaluation of the kinematics at random test points, and validation of arbitrary trajectory inputs. The results of all experiments compared the analytical solutions derived from the TCM [1] (black), the actual robot positions measured with an electromagnetic sensor (red), and the predicted positions from the proposed kinematic model (blue), as shown in Fig. 10.

To capture nonlinearity and snapping, a full rotation was performed for $\beta_{12} = 15$, 25, 35, 45, and 55, as shown in Fig. 10(a). As β_{12} increased, the TCM struggled to accurately predict forward kinematics. The transformation matrix was used to map the coordinates between the TCM and the learning-based kinematic model. For smaller β_{12} values, the proposed model accurately captured minor misalignments caused by hardware failure. Although mathematical models can be sufficient under ideal conditions, the proposed model addresses the practical realities of non-ideal robots, including manufacturing imperfections, defects, and unmodeled factors. At $\beta_{12} = 45$, the TCM predicted nonlinearity but failed at $\beta_{12} = 55$, where snapping occurred. In contrast, the proposed model predicted the snapping point and its magnitude. During snapping, the proposed model exhibited position and orientation errors of 0.8 mm and 4.2°, respectively, except at the snapping point, where errors increased to 12 mm and 20°. Larger errors shortly occur at the snapping point due to a 1–2 step delay introduced by the input buffer, but they quickly diminish. This indicates that the model predicts snapping timing and magnitude more accurately than the TCM.

The second experiment evaluated predictions for 20 random points being excluded from the training data to assess the model's performance at specific locations or configurations in the spatial domain. Fig. 10(b) shows that the theoretical values closely matched the actual values in most cases; however, substantial errors were observed at specific points, such as at points 8, 9, and 13. The TCM exhibited errors of 8.55 mm and 7.88°, while the learning model achieved significantly lower errors of 0.87 mm and 3.00°. In terms of computation time, the TCM required 1.47 ± 2.930 s per point, whereas the learning model required only 0.831 ± 0.200 ms, which is suitable for real-time control applications.

The third experiment evaluated the predictions for the arbitrary inputs. The arbitrary joint inputs comprised the outer tube translating β_2 from 20 to 15 mm and rotating α_2 from 0°

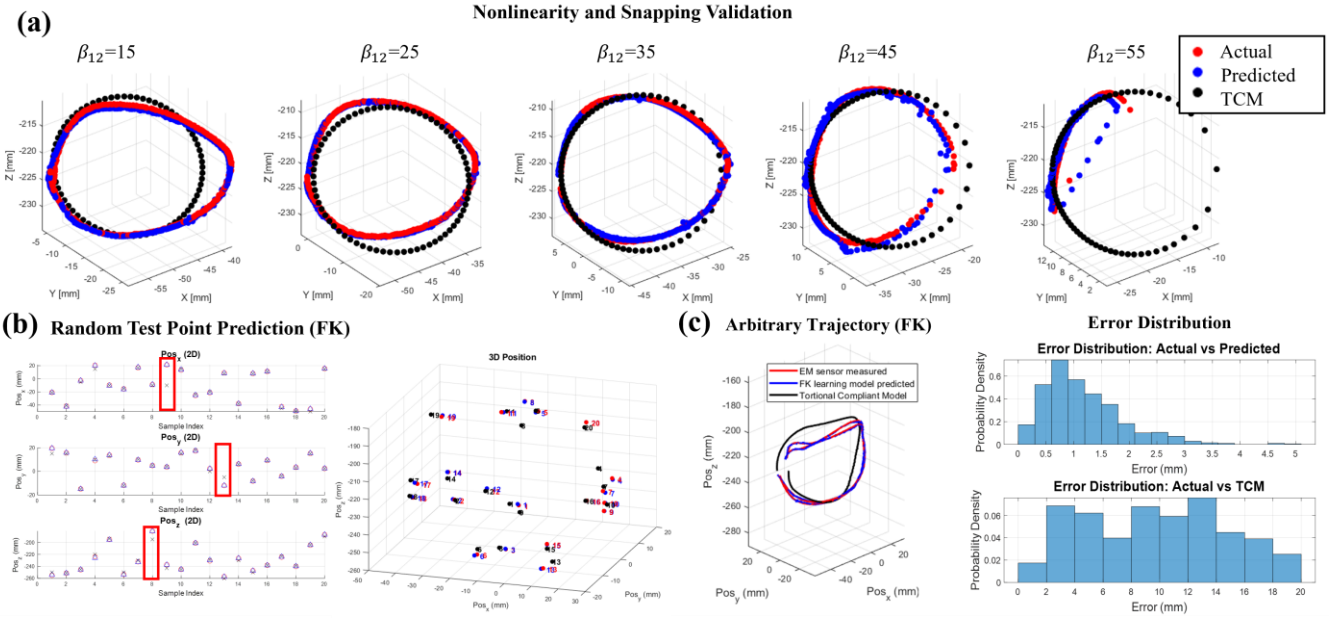

Fig. 10. Forward kinematics validation. (a) Nonlinearity and snapping validation. Black, blue, and red points denote TCM, model predictions, and real tip pose, respectively. (b) Random 20 points prediction result. (c) Arbitrary trajectory validation and error distribution.

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

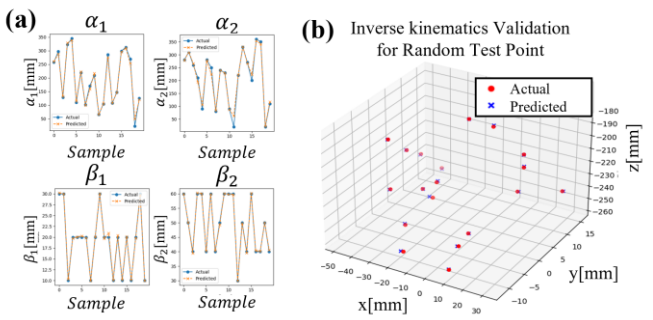

Fig. 11. Inverse kinematics validation with random points. (a) Joint values and (b) comparison of given and predicted points though forward kinematics.

to 360°. Simultaneously, the inner tube translated β_1 from 30 mm to 40 mm with the rotation of α_1 from 360° to 0°. As shown in Fig. 10(c), the TCM errors were 10.94 mm and 11.11°, whereas the proposed model errors were 1.37 mm and 4.87°. The proposed model performed remarkably in closely trac king small path deviations caused by directional changes, indicating its ability to account for the effects of history on directional parameters. The error density analysis demonstrated that the learning-based kinematic model consistently maintained errors of approximately 1 mm, while the TCM exhibited a broader error distribution, with errors reaching up to 20 mm, as shown in Fig. 10(c).

B. Inverse Kinematics

When applying the conventional approach outlined in [28] to our robot, the computation time for the inverse kinematics ranged from 0.2 s to 8 s for a single point. In some cases, the solution failed to converge within the given time frame, making this approach unsuitable for real time control and teleoperation. To address these limitations, a learning-based, lightweight model that was entirely independent of mathematical modeling was developed.

After training the model by mapping the tip poses and joint angles, it was validated as follows. In the first experiment, 20 random tip poses were selected to calculate joint values using the trained inverse kinematics model. Given that the original joint values were known, the accuracy of the predictions can be verified. The predicted joint values are shown in Fig. 11(a). The predicted joint values were then input into the forward kinematics model described in Section IV.A to compute the robot tip positions in 3D space. The computed positions were compared to the previous 20 random tip poses, as shown in Fig. 11(b). All points were predicted accurately, and the distance error between the restored tip positions and the given random points was less than 1 mm. The computation time of calculating a single point of the inverse kinematics was 0.816±0.200ms. The prediction error of the joint values was 1.22 mm and 2.46°. These results confirm the efficiency and precision of the proposed lightweight system, ensuring its suitability for real-time microsurgical applications.

In the second experiment, arbitrary paths in 3D space were provided, and the inverse kinematics model was used to predict the joint values required to follow these paths. Given that the original joint values were unknown, the predicted joint values

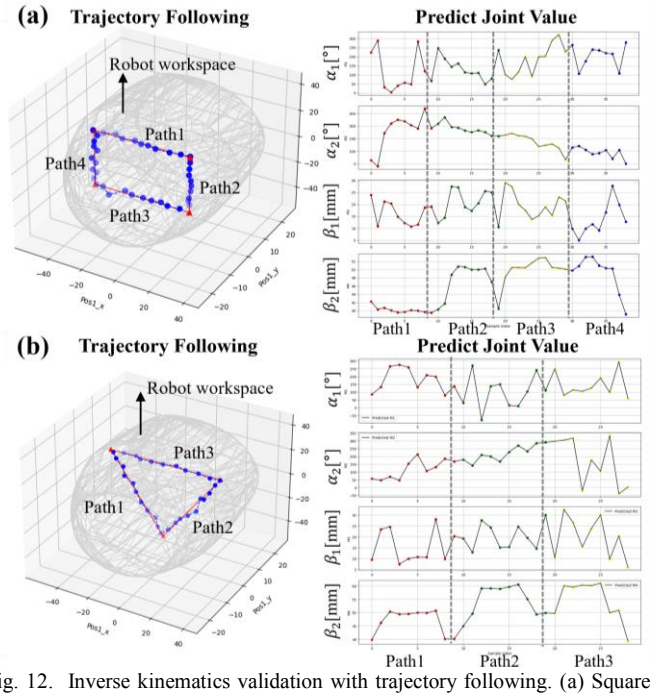

Fig. 12. Inverse kinematics validation with trajectory following. (a) Square path and predicted joint value. (b) Triangular path and predicted joint value.

were input into the forward kinematics model and compared with the given paths. Two distinct path shapes—square and triangular—were used in this experiment. As shown in Fig. 12, both paths were carefully designed to fit within the robot's entire workspace, ensuring that all predicted joint values corresponded to feasible configurations for the robot. Each path was divided into 10 points to interpolate the given trajectory. For orientation interpolation, its rotational matrices were converted u sing the Equivalent Angle Axis method, the orientation was divided into 10 segments, then the intermediated values were converted into the rotational matrices back. The results shown in Fig. 12(a) and Fig. 12(b) reveal that despite occasional discontinuities in the predicted joint values, the forward kinematics confirmed that all points closely followed the given trajectory. In Fig. 12(a), the errors were 2.21 mm and 5.88 degrees, while in Fig. 12(b), they were 2.95 mm and 7.90 degrees. Most predictions were accurate within 1 mm and 3 degrees, but larger errors arose for physically unreachable target points or untrained regions. This highlights the issue of multiplicity, which will be addressed in future path-planning research.

C. Discussion

This study provides valuable insights into the challenges and opportunities of using a learning-based CTR kinematic model. However, it has several limitations. Unlike previous studies [16, 18, 19] that used three-tube CTRs, our model was developed and validated for a two-tube system, effectively capturing its nonlinear behaviors. Nevertheless, additional validation is required to extend the approach to more complex nonlinearities observed in three-tube CTRs. Furthermore, uncertain predictions may occur in untrained regions, and the issue of multiplicity may arise which means that the deep learning model can generate solutions even when the desired input

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

corresponds to physically infeasible robot postures. When the robot's position or posture in inverse kinematics is outside the workspace, the model may return invalid outputs, potentially compromising microsurgical precision and safety. Although the input buffer-based method significantly improves accuracy compared to the traditional TCM, slight errors may still occur momentarily at snapping points. Therefore, incorporating more refined constraints during control is necessary for practical implementation. Additionally, for path planning in vitreoretinal surgery, clinical evaluation of the required motions should be conducted. Despite these limitations, our study provides a foundation for learning-based approaches to address nonlinear CTR kinematics. If additional methods for path planning —either avoiding or incorporating snapping—are developed, this research will contribute to broader adoption of CTR in minimally invasive surgeries, including vitreoretinal surgery.

V. Conclusion

This letter proposes a learning-based kinematic model for CTR that considers history dependence and nonlinearity while effectively capturing snapping and hysteresis behaviors. The proposed model achieved errors of 0.69 mm and 0.16° in forward kinematics and 1.22 mm and 2.46° in inverse kinematics, outperforming existing methods [16, 18, 19]. Additionally, the lightweight model, with a computation time of 0.8 ms, enabled real-time control while accurately modeling extreme motions. Future efforts will focus on designing a robust system for stable real-time teleoperation control, addressing the limitations of traditional mathematical models and hardware to enhance precision in microsurgical procedures.

References

[1] P. E. Dupont, J. Lock, B. Itkowitz and E. Butler, "Design and Control of Concentric-Tube Robots," in *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 209-225, April 2010.

[2] J. Burgner-Kahrs, D. C. Rucker and H. Choset, "Continuum Robots for Medical Applications: A Survey," in *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1261-1280, Dec. 2015.

[3] J. Ha, G. Fagogenis and P. E. Dupont, "Modeling Tube Clearance and Bounding the Effect of Friction in Concentric Tube Robot Kinematics," in *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 353-370, April 2019.

[4] P. Sears and P. E. Dupont, "Inverse Kinematics of Concentric Tube Steerable Needles," *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Rome, Italy, pp. 1887-1892, 2007.

[5] D. C. Rucker, B. A. Jones and R. J. Webster III, "A Geometrically Exact Model for Externally Loaded Concentric-Tube Continuum Robots," in *IEEE Transactions on Robotics*, vol. 26, no. 5, pp. 769-780, Oct. 2010.

[6] R. J. Hendrick, H. B. Gilbert and R. J. Webster, "Designing snap-free concentric tube robots: A local bifurcation approach," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, Seattle, WA, USA, pp. 2256-2263, 2015.

[7] J. -S. Kim, D. -Y. Lee, K. Kim, S. Kang and K. J. Cho, "Toward a solution to the snapping problem in a concentric-tube continuum robot: Grooved tubes with anisotropy," *2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, pp. 5871-5876, 2014.

[8] H. Azimian, P. Francis, T. Looi and J. Drake, "Structurally-redesigned concentric-tube manipulators with improved stability," *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Chicago, IL, USA, pp. 2030-2035, 2014.

[9] C. Bergeles, A. H. Gosline, N. V. Vasilyev, P. J. Codd, P. J. del Nido and P. E. Dupont, "Concentric Tube Robot Design and Optimization Based on Task and Anatomical Constraints," in *IEEE Transactions on Robotics*, vol. 31, no. 1, pp. 67-84, Feb. 2015.

[10] C. Greiner-Petter and T. Sattel, "On the influence of pseudoelastic material behavior in planar shape-memory tubular continuum structures" , Smart Materials and Structures, vol.26, no.12, pp.125024, 2017.

[11] M. Khadem, J. O'Neill, Z. Mitros, L. d. Cruz and C. Bergeles, "Autonomous Steering of Concentric Tube Robots for Enhanced Force/Velocity Manipulability," *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, pp. 2197-2204, 2019.

[12] M. Khadem, J. O'Neill, Z. Mitros, L. da Cruz and C. Bergeles, "Autonomous Steering of Concentric Tube Robots via Nonlinear Model Predictive Control," in *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1595-1602, Oct. 2020.

[13] K. Iyengar, S. M. H. Sadati, C. Bergeles, S. Spurgeon and D. Stoyanov, "Sim2Real Transfer of Reinforcement Learning for Concentric Tube Robots," in *IEEE Robotics and Automation Letters*, vol. 8, no. 10, pp. 6147-6154, Oct. 2023.

[14] K. Iyengar, S. Spurgeon and D. Stoyanov, "Deep Reinforcement Learning for Concentric Tube Robot Path Following," in *IEEE Transactions on Medical Robotics and Bionics*, vol. 6, no. 1, pp. 18-29, Feb. 2024.

[15] A. Kuntz, A. Sethi and R. Alterovitz, "Estimating the complete shape of concentric tube robots via learning", *Proc. Hamlyn Symp. Med. Robot.*, pp. 43-44, Jun. 2019.

[16] G. Fagogenis, C. Bergeles and P. E. Dupont, "Adaptive nonparametric kinematic modeling of concentric tube robots," *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea (South), pp. 4324-4329, 2016.

[17] A. Kuntz, A. Sethi, R. J. Webster and R. Alterovitz, "Learning the Complete Shape of Concentric Tube Robots," in *IEEE Transactions on Medical Robotics and Bionics*, vol. 2, no. 2, pp. 140-147, May 2020.

[18] R. Grassmann, V. Modes and J. Burgner-Kahrs, "Learning the Forward and Inverse Kinematics of a 6-DOF Concentric Tube Continuum Robot in SE(3)," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, pp. 5125-5132, 2018.

[19] N. Liang, R. M. Grassmann, S. Lilge and J. Burgner-Kahrs, "Learning-based Inverse Kinematics from Shape as Input for Concentric Tube Continuum Robots," *2021 IEEE International Conference on Robotics and Automation (ICRA)*, Xi'an, China, pp. 1387-1393, 2021.

[20] S. M. H. Sadati, Z. Mitros, R. Henry, L. Zeng, L. d. Cruz and C. Bergeles, "Real-Time Dynamics of Concentric Tube Robots With Reduced-Order Kinematics Based on Shape Interpolation," in *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5671-5678, April 2022.

[21] Y. Wang *et al*., "Using Neural Networks to Model Hysteretic Kinematics in Tendon-Actuated Continuum Robots," *2024 International Symposium on Medical Robotics (ISMR)*, Atlanta, GA, USA, pp. 1-7, 2024.

[22] W. Bai *et al*., "Task-Based LSTM Kinematic Modeling for a Tendon-Driven Flexible Surgical Robot," in *IEEE Transactions on Medical Robotics and Bionics*, vol. 4, no. 2, pp. 339-342, May 2022.

[23] B. Xiao *et al*., "Learning-Based Inverse Kinematics Identification of the Tendon-Driven Robotic Manipulator for Minimally Invasive Surgery," *IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society*, Singapore, Singapore, pp. 1-6, 2023.

[24] G. Fang, Y. Tian, Z. -X. Yang, J. M. P. Geraedts and C. C. L. Wang, "Efficient Jacobian-Based Inverse Kinematics With Sim-to-Real Transfer of Soft Robots by Learning," in *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 5296-5306, Dec. 2022.

[25] G. Jeong and S.Y. Ko, "A Novel Vitreoretinal Surgical Robot System to Maximize the Internal Reachable Workspace and Minimize the External Link Motion", 2024 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Abu Dhabi, UAE, 2024.

[26] R. M. Grassmann, R. Z. Chen, N. Liang and J. Burgner-Kahrs, "A Dataset and Benchmark for Learning the Kinematics of Concentric Tube Continuum Robots," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Kyoto, Japan, pp. 9550-9557, 2022.

[27] D. Kim, H. Kim and S. Jin, "Recurrent Neural Network With Preisach Model for Configuration-Specific Hysteresis Modeling of Tendon-Sheath Mechanism," in *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2763-2770, April 2022.

[28] M. Khadem, L. Da Cruz and C. Bergeles, "Force/Velocity Manipulability Analysis for 3D Continuum Robots," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, pp. 4920-4926, 2018.