Estimating the Complete Shape of Concentric Tube Robots via Learning

A. Kuntz, A. Sethi, R. Alterovitz

Department of Computer Science, University of North Carolina at Chapel Hill adkuntz@cs.unc.edu (DOI10.31256/HSMR2019.22)

INTRODUCTION

Concentric tube robots, needle diameter robots composed of nested pre-curved tubes, are capable of curving around anatomical obstacles to perform surgical procedures at difficult-to-reach sites. Concentric tube robots have the potential to enable less invasive surgeries in many areas of the human body, including the skull base, the lungs, and the heart [1]. For these robots, in order to safely control and plan motions that automatically prevent unintended collisions with the patient's anatomy, an accurate shape model of the entire robot's shaft is required. Accurate prediction of the entire shape of a concentric tube robot from its control inputs is challenging, and current state-of-the-art shape models are often unable to accurately account for complex and unpredictable physical phenomena such as inconsistent friction between tubes, non-homogenous material properties, and imprecisely shaped tubes [2][3]. In this work, we present a data driven, deep neural network-based approach for learning a more accurate model of a concentric tube robot's entire shape.

Machine learning enables a data driven approach to the shape estimation of concentric tube robots. Neural network models have been successfully used to more accurately model the forward kinematics and inverse kinematics of concentric tube robots [4][5], and an ensemble method has been applied to learn and adapt a forward kinematics model online [3]. However, these models only consider the pose of the robot's tip. In order to successfully plan and execute motions that avoid unwanted collisions between the robot's shaft and patient anatomy, a model must accurately predict the *entire shape* of the robot.

In this work, we present a deep neural network approach that learns a function that accurately models the entire shape of the concentric tube robot, for a given set of tubes, as a function of its configuration (see Fig. 1). The neural network takes as input the robot's configuration, and the network outputs coefficients for orthonormal polynomial basis functions in x, y, and z parameterized by arc length along the robot's tubular shaft. In this way, a function representing the entire shape of the robot can be produced by one feed forward pass through the neural network.

The key insight behind our parameterization is that the uncertainty in the physics-based shape models is due mainly to uncertainty in curvature and torsion. The *arc length* of the robot's shape, however, is independent of these and as such is generally not subject to the same sources of uncertainty. We can leverage this known state by parameterizing our shape function by arc length.

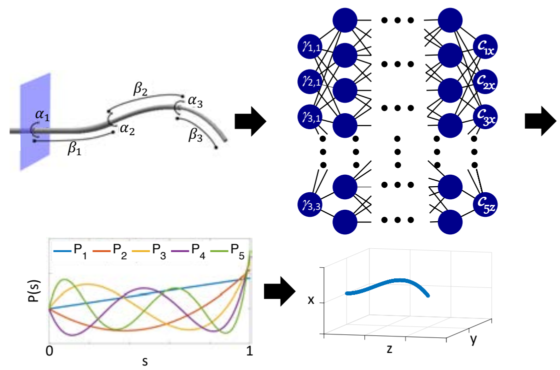

Fig. 1 Given a concentric tube robot configuration defined by the translations and rotations of the tubes (upper left), our neural network (upper right) outputs coefficients for a set of polynomial basis functions (lower left) that are combined to model the backbone of the robot's 3D shape (lower right).

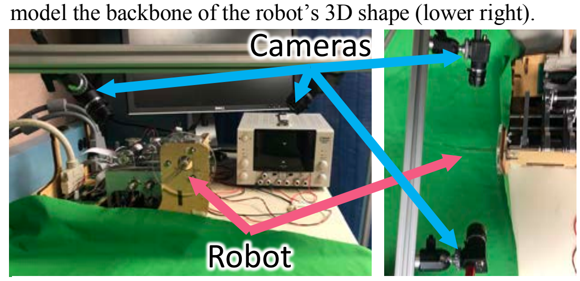

Fig. 2 We train the neural network using data from a physical robot. By taking images from multiple cameras (blue arrows), the shape of the robot's shaft (pink arrows) can be reconstructed in 3D using shape from silhouette.

MATERIALS AND METHODS

In order to learn a shape function for the concentric tube robot, data representing the robot's shape as a function of its configuration must be gathered. To gather shape data, we utilize a multi-view 3D computer vision technique called shape from silhouette [6], in which multiple images of the robot's shape for a given configuration are collected from cameras with known position (see Fig. 2). The robot's shaft is then segmented in each image and for each pixel in the segmentation a ray is traced out from the camera's position through its image plane. These rays then pass through a voxelized representation of the world, and voxels that are intersected by rays from every camera represent the robot's shape in the world. We then fit a 3D space curve to the voxels using ordinary least squares, resulting in a curve that represents the true, sensed backbone of the robot. We then train the neural network using the sensed backbone as ground truth.

Our neural network architecture consists of a feed forward, fully connected network, with 5 hidden layers of 30 nodes each. We utilize the parametric rectified linear unit as our non-linear activation function between layers, which we noted provided a slight performance improvement over the standard rectified linear unit.

For a robot consisting of k tubes, we parameterize the i'th tube's state as $\boldsymbol{\gamma}_i := \{\gamma_{1,i}, \gamma_{2,i}, \gamma_{3,i}\} = \{\cos(\alpha_i), \sin(\alpha_i), \beta_i\}$ where $\alpha_i \in (-\pi, \pi]$ is the i'th tube's rotation and $\beta_i \in \mathbb{R}$ is the i'th tube's translation, as in [5]. We then parameterize the robot's configuration as $\mathbf{q} = (\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_k)$. This serves as the input to the neural network.

The network outputs 15 coefficients, c_{1x}, c_{2x}, … c_{5x}, c_{1y}, c_{2y}, … c_{5y}, c_{1z}, c_{2z}, … c_{5z}, which serve as coefficients for a set of 5 orthonormal polynomial basis functions in x, y, and z parameterized by arc length, shown in Table 1. This results in three functions, $x(\mathbf{q}, s)$, $y(\mathbf{q}, s)$, and $z(\mathbf{q}, s)$, where $x(\mathbf{q}, s) = \mathrm{len}(\mathbf{q}) * (c_{1x}\mathrm{P}_1(s) + c_{2x}\mathrm{P}_2(s) + \ldots + c_{5x}\mathrm{P}_5(s))$, where s is a normalized arc length parameter between 0 and 1, and len($\mathbf{q}$) is the total arc length of the robot's backbone in configuration $\mathbf{q}$. Then $y(\mathbf{q}, s)$ and $z(\mathbf{q}, s)$ are defined similarly with their respective coefficients. The resulting shape function is $\mathtt{shape}(\mathbf{q}, s) = \langle x(\mathbf{q}, s), y(\mathbf{q}, s), z(\mathbf{q}, s)\rangle$. To evaluate the shape of the robot at a given configuration, the neural network can be evaluated at $\mathbf{q}$, and the resulting coefficients define a space-curve function that can then be evaluated at any desired arc length. This, combined with knowledge of the robot's radius as a function of arc length, results in a prediction of the robot's geometry in the world.

We first pretrained our model on 100,000 data points (configuration and backbone pairs) generated by the physics-based model presented in [2]. Such pretraining allows us to prevent overfitting on the smaller amount of sensed, real world data.

We then trained our network on 8,000 data points, and we evaluate the network on 1,000 different test data points (both sets generated via shape from silhouette). We utilize a pointwise sum-of-squared-distances loss function and the ADAM [7] optimizer during training.

	s	s^2	s^3	s^4	s^5
$\mathrm{P}_1(s)$	1.7321	0	0	0	0
$\mathrm{P}_2(s)$	-6.7082	8.9943	0	0	0
$\mathrm{P}_3(s)$	15.8745	-52.915	39.6863	0	0
$\mathrm{P}_4(s)$	-30.0	180.0	-315.0	168.0	0
$\mathrm{P}_5(s)$	49.7494	-464.33	1392.98	-1671.6	696.4912

Table 1 Coefficients for the orthonormal polynomial basis functions generated using Gram-Schmidt orthogonalization. $\mathrm{P}_1(s) := 1.7321s$, $\mathrm{P}_2(s) := -6.7082s + 8.9943s^2$, etc., plotted in Fig. 1 (lower left).

RESULTS

We compare our neural network's shape computation to that of the physics-based model presented in [2]. In Fig. 3 we plot a histogram of the errors across the 1,000 test configurations. For each configuration of our 3-tube robot we evaluate the shape of the physics-based model, the learned model, and the ground truth from the vision system at 20 evenly spaced points along the robot's shaft. We then present the maximum error—the L_2 distance of the point that deviates from the ground truth the most. The error distribution of the learned model is shifted to the left compared with that of the physics-based model, indicating that the learned model is more likely to produce lower error values than the physics-based model.

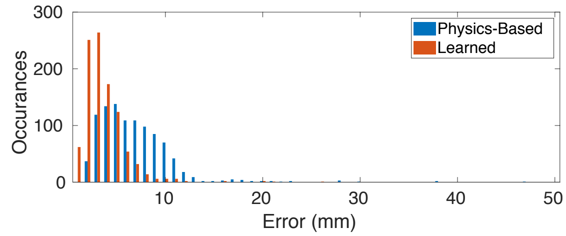

Fig. 3 A histogram of the maximum error along the robot's shaft for the learned model and the physics-based model, for each of the 1,000 test points. The distribution is shifted to the left in the learned model, indicating that it is more likely to produce lower error values. The maximum error is 26mm for the learned model and 47mm for the physics-based model.

DISCUSSION

In this work we present a learned, neural network model that outputs an arc length parameterized space curve. This allows us to take a data driven approach to modeling the shape of the concentric tube robot and improve upon a physics-based model. This may allow for safer motion planning and control of these devices in surgical settings that require avoiding anatomical obstacles. The model is only trained on cases where the robot is operating in free space. Accounting for interaction with tissue is the subject of future work. We also intend to investigate other models and augment the learned model to account for other sources of uncertainty in concentric tube robot shape modeling, including hysteresis, and plan to integrate the learned model with a motion planner and evaluate its use in automatic obstacle avoidance during tele-operation or automatic execution of surgical tasks.

ACKNOWLEDGEMENTS

This research was supported in part by the National Institutes of Health under Award R01EB024864.

REFERENCES

[1] Burgner-Kahrs J., Rucker D.C., Choset H. Continuum robots for medical applications: a survey. IEEE Trans. Robotics 2015 31(6):1261-80.
[2] Rucker D.C. The mechanics of continuum robots: model-based sensing and control. Ph.D. dissertation, Vanderbilt University, 2011.
[3] Fagogenis G., Bergeles C., Dupont P.E. Adaptive nonparametric kinematic modeling of concentric tube robots. IEEE/RSJ Int. Conf. on Int. Robots and Systems (IROS), Oct. 2016, 4324-29.
[4] Bergeles C., Lin F.Y., Yang G.Z. Concentric tube robot kinematics using neural networks. Hamlyn Symp. on Med. Robotics, 2015, 1-2.
[5] Grassmann R., Modes V., Burgner-Kahrs J. Learning the forward and inverse kinematics of a 6-DOF concentric tube continuum robot in SE(3). IEEE/RSJ Int. Conf. on Int. Robots and Systems (IROS), Oct. 2018, 5125-32.
[6] Baker S., Kanade T. Shape-from-silhouette across time part I: Theory and algorithms. Int. Journal of Comp. Vis. 2005 62(3):221-47.
[7] Kingma D.P., Ba, J. Adam: A method for stochastic optimization. arXiv preprint, 2014. 1412.6980.