Sim-to-Real Transfer of Soft Robotic Navigation Strategies That Learns from the Virtual Eye-in-Hand Vision

Jiewen Lai, Tian-Ao Ren, Wenchao Yue, Shijian Su, Jason Y. K. Chan, Hongliang Ren

Sim-to-Real Transfer of Soft Robotic Navigation Strategies That Learns from the Virtual Eye-in-Hand Vision

Jiewen Lai†, Tian-Ao Ren†, Wenchao Yue, Shijian Su, Jason Y. K. Chan, and Hongliang Ren

***Abstract*— To steer a soft robot precisely in an unconstructed environment with minimal collision remains an open challenge for soft robots. When the environments are unknown, prior motion planning for navigation may not be available for both simulation and operation. This paper presents a novel Sim-to-Real method to guide a cable-driven soft robot in a static environment under the Simulation Open Framework Architecture (SOFA). The scenario aims to resemble one of the steps during a simplified transoral tracheal intubation process where a robotic endotracheal tube is guided to the upper trachea-larynx location by a flexible video-assisted endoscope/stylet. In SOFA, we employ the quadratic programming inverse solver to obtain collision-free motion strategies for the endoscope/stylet manipulation based on the robot model and encode the virtual eye-in-hand vision. Then, we associate the anatomical features recognized by the virtual vision and the joint space motion using a closed-loop nonlinear autoregressive exogenous model (NARX) network. Afterward, we transfer the learned knowledge to the robot prototype, expecting it to navigate to the desired spot in a new phantom environment automatically based on its eye-in-hand vision only. Experiment results indicate that our soft robot can efficaciously navigate through the unstructured phantom to the desired spot with minimal collision motion according to what it has learned from the virtual environment. The results show that the average R-squared coefficient between the closed-loop NARX-forecasted and SOFA-referenced robot's cable and prismatic joint space motion are 0.963 and 0.997, respectively. The eye-in-hand visions also demonstrate good alignment between the robot tip and the glottis.**

***Index Terms*— Soft Robotics, Robot Learning, Motion Planning, Simulation.**

I. Introduction

This work was supported in part by the Hong Kong Research Grants Council (RGC) Collaborative Research Fund (CRF-C4026-21GF); and in part by the Hong Kong Research Grants Council (RGC) Research Impact Fund (RIF-R4020-22); and in part by the Dr. Barbara Kwok Young Postdoctoral Fellow Travel Grants Award. (*Corresponding Author: Hongliang Ren*)

J. Lai, W. Yue, S. Su, and H. Ren are with the Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong, China (e-mail: hlren@ee.cuhk.edu.hk)

T.-A. Ren is with the College of Mechanical and Electrical Engineering, Beijing University of Chemical Technology, Beijing, China; he is also with the CUHK Shenzhen Research Institute (SZRI), Shenzhen, China.

J. Y. K. Chan is with the Department of Otorhinolaryngology, Head and Neck Surgery, The Chinese University of Hong Kong, Shatin, Hong Kong, China.

†These two authors contributed equally to this work.

This article has a supplementary video provided by the authors.

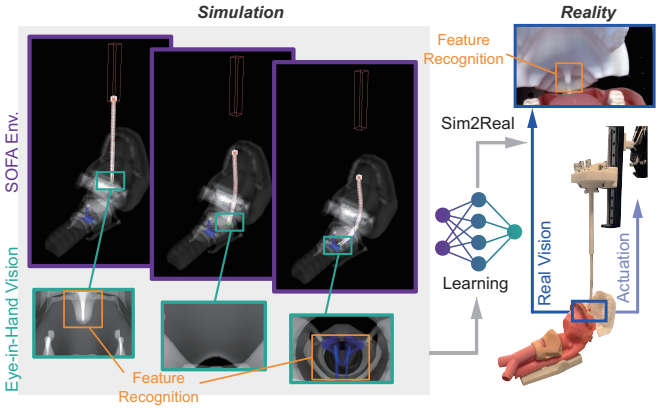

Fig. 1. Schematic diagram: Transferring the learned policies that actuate the soft robot with an optimal motion from the SOFA environment to a real-world system based on virtual and real eye-in-hand vision.

MANY soft robot manipulators and systems have been designed and intended for the applications of medical intervention in the past few decades [1]. They are ideal candidates for robotic surgical tools when force transmission is a noncritical factor [2]. Inspired by biological compliant structures, these soft continuum robots can navigate or work in complex environments with the employment of well-established kinematics, dynamics, and mechanics [3]–[5]. Besides, due to the challenges in describing those highly nonlinear compliant manipulators made from soft materials with low Young's Modulus and interaction [6], model-free approaches like visual servoing [7], sensorimotor learning [8], and finite element methods (FEM) [9] were widely utilized in the soft/continuum robotic control. In general, a flexible robotic medical intervention would request a 2D/3D reconstruction of the device in an occlusive environment. The used-to-be difficult reconstruction is now becoming convenient because of the technological advances in sensors, such as electromagnetic sensing [10], Fiber Bragg grating (FBG) [11], and learning-based strain gauges-liked networks [8], but at a relatively high cost. In addition, prototype-dependent sensory systems often require re-calibration on every individual device, as the multi-sensor assembly may differ from one to another. For example, a learning-based embodied strain gauges array system may become invalid when deployed to another identical flexible robotic system due to minor assembly errors in

the real world, necessitating local black-box re-learning. We expect that an approach with a minimal amount of sensors and on-site calibration could significantly generalize these novel soft devices to practical use with lower cost and higher reproductivity.

Simulation-to-Reality ("Sim-to-Real" or "Sim2Real") transfer learning will meet our claimed expectations. Bonding the linkage between simulation and reality is one of the essential steps toward the metaverse. With extensive prior study in simulation, one can reproduce the virtual result on a physical soft robot. For example, with extensive studies in kinematics, dynamics, mechanics, and morphology, model-based simulators can be developed to contribute to different control problems such as contact detection [12], soft materials shrinkage upon actuation [13], hybrid rigid-soft robots [14], soft parallel robots [15], etc. In addition to robot control, simulators can sometimes help us to design better soft robots to fit different applications [16]–[18].

Another stream of soft robotic simulations may include the virtual world's sensing, and the physical interaction [19]. A cohort of researchers from French institutes developed a soft robots plugin for the Simulation Open Framework Architecture (SOFA) [20], [21], with physics-based soft body dynamics. The soft robots plugin is capable of deriving the quantitative relationship between the robot's deformation and the changes in the inputs of the actuators (i.e., joint space) based on a real-time direct/inverse FEM solver that considers mechanical parameters like material, geometry, and morphology. Such an open-source toolkit has been useful for the community to probe into the robots' modeling, characterization, and interaction problems with plausible visualization before or during the transfer to prototypes.

However, most of the reported applications are solely for visualization without online deployment. In fact, a reliable simulation can be utilized to advise a closed-loop control strategy based on the robots' perception in the virtual environment. By using different simulation techniques, sim-to-real transfer learning was applicable to industrial robots. For instance, [22] presents a sim-to-real learning method that trains a rigid manipulator in MuJoCo to avoid colliding with obstacles and then transfers to the physical world using 3D bounding boxes estimated from RGB-D vision. In [23], a sim-to-real transfer method is introduced for reinforcement learning deployed on a KUKA LBR iiwa arm for a peg-in-hole task with PyBullet. Due to the availability of the well-developed simulation platform and mature robot models in Unified Robotics Description Format (URDF), perception beyond joint space is no longer a must in the closed-loop feedback. As a result, the learned policy of rigid robots is oftentimes and readily applicable in the real world. While sim-to-real-based control policies are common in rigid robots (e.g., CoppeliaSim, MuJoCo, and RoboDK, to name a few), they have rarely been reported on soft robots until recent years.

Soft robot-wise, sim-to-real transfer methods can assist the robots design and fabrication [24]. The calibration of vision-based 3D shape sensing of pneumatic soft robots can also be trained in simulation and transferred to real-world deployment [25]. In [26], an open-source sim-to-real transfer method is put forward to predict the morphology of cube-based soft robotic dice. The work is further extended to transfer the simulated locomotion to reality [17]. By exploring planar kinematics, which can be geometrically simulated, [27] presents the autonomous navigation task for soft growing robots in a tortuous maze with an overhead view. However, 3D navigation, a capability of soft manipulators that are often sought after and competent in and which could have benefited from the sim-to-real transfer, has yet to be reported.

This work proposes a SOFA-based sim-to-real method for soft robotic navigation that learns from virtual eye-in-hand vision. Based on the underlying principle of the simulator, we assume that the SOFA's results may be very likely to resemble the real case scenarios. Aiming to navigate a cable-driven soft robot in a confined environment, we first reconstruct a simulation scene in the SOFA framework that resembles the situation to perform collision-free path and motion planning that could be useful for endoscopic manipulation. Then, we employ the prior knowledge from the simulation to train a closed-loop control policy for a soft robot's navigation. By learning what the robot "sees" and how it simultaneously "moves" in the joint space according to a series of optimized motions in the virtual world, we can transfer the learned policy to the physical robot and teach it how to "move" depending on what it "sees" in the real world – and the environment is unknown to the robot except for regular anatomical features we are intrigued in. A dynamic neural network called nonlinear auto-regressive exogenous model (NARX) [28] is adopted for virtual learning that features time-series modeling. Instead of presenting a simple open-loop sim-to-real method, our closed-loop policy can be directly transferred to the tangible robot in a one-off manner, which can considerably reduce the on-site calibration and multisensory employment for the navigation tasks. Experiments with further evaluation validate the method's feasibility and performance. This paper contributes to the soft and medical robot communities

- A newly-presented 3D-printed cable-driven soft robotic system featuring a miniature manipulator, soft material, and mechatronic-decoupled design for soft robot-based endoscopic manipulation;
- A novel SOFA-based sim-to-real method that learns from the virtual eye-in-hand vision for simulated and real-world soft robotic navigation relying on a light-weighted NARX network;
- An interdisciplinary pilot study of autonomous soft robot-based endoscopic manipulation powered by our sim-to-real method; and
- A comprehensive experimental validation and evaluation of our sim-to-real method for soft robots.

To the best of our knowledge, this is the first physical simulation-based sim-to-real method that enables soft robotic navigation that learns from virtual eye-in-hand vision. The method enables the transfer of complicated soft robot motion computed by a numerical solver in SOFA to a real-world robot with additional visual perception to improve transfer fidelity.

This paper is structured as follows: Sec. II describes the gist of (soft) robotic transoral tracheal intubation with its

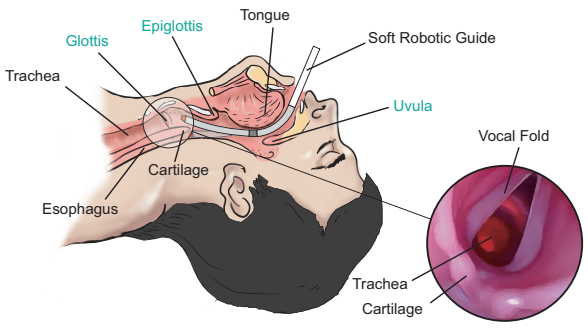

Fig. 2. Task description: the robot tip is automatically steered to reach the upper glottis with the help of the selected anatomical features obtained by its eye-in-hand vision. The soft body's motion with a minimal collision with the surrounding is realized by the control strategy that learns from the simulation.

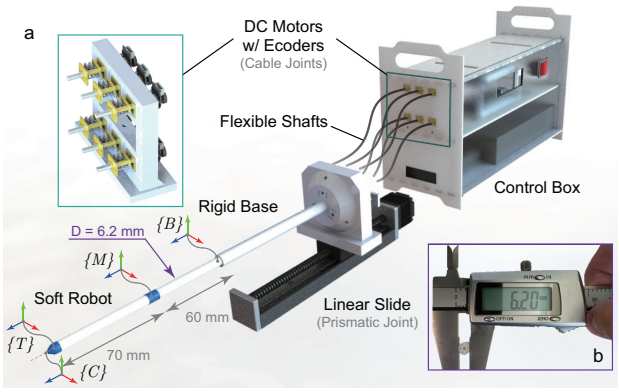

Fig. 3. (a) CAD schematic of the soft robot system in this work. A total of six cable joints are enabled by the motors' rotation that is transmitted by the flexible shafts. (b) The outer diameter of the soft robot is 6.2 mm. Notation of frames: $\{B\}$: base; $\{M\}$: middle; $\{T\}$: tip; $\{C\}$: camera.

background introduction. Sec. III sketches out the design and assembly of the robotic system that will be used in simulation and experiment. Sec. IV demonstrates the kernels of how we construct the soft robotic-based endoscope (stylet) manipulation scene, motion planning, and generation of the dataset in SOFA. Sec. V presents the implementation of learning. Sec. VI shows the experimental validation of the proposed method on a phantom. The last section concludes the paper.

II. Task Description

We assume the primary task for the soft robot is that it can automatically guide the soft robot's tip to a reachable 3D spot in a confined environment with optimal body motion throughout the navigation process. Here, we choose the robotic endoscope/stylet manipulation in transoral tracheal intubation (TI) as an example to investigate the feasibility and performance of our proposed sim-to-real method. A stylet or flexible bronchoscope is typically used to guide the endotracheal tube (ETT) to reach the desired spot. Despite the conventional "blind" stylets, video-assisted [29] and semi-robotic stylets [30] were proposed to help with the transoral TI. However, it comes to our attention that, except for pink tissues, the endoscope fails to provide identifiable views for a good while during the navigation. The situation poses major challenge in deploying visual servoing for the task. Nonetheless, we see the potential to overcome the challenge by using a soft robot with sim-to-real capability.

To automate this procedure, as illustrated by Fig. 2, we assumed that a vision-embedded soft robotic manipulator would work as a steerable stylet to autonomously navigate to the locations near the upper glottis with minimal robot-environment collisions during the feeding. Along with feeding, there are two major turnings for the soft body. The first major turning, obviously, occurs near the palatine uvula that separates the oral cavity and oropharynx. After a blackout period when no key features can be perceived, the second turning occurs near the arytenoid (corniculate) cartilage at the hypo-pharyngeal area separating the trachea and esophagus [31]: the former belongs to the respiratory system, whereas the latter belongs to the digestive system. We can employ the null space motion of a multisegment soft robot with proper motion planning to produce a dexterous motion that avoids collisions as much as possible.

III. Soft Robotic System: Design & Assembly

A soft robotic system and its manipulator parameters were specially designed to validate our method. As shown in Fig. 3, the cable-driven soft robot has two coupled flexible segments. Three independent cables actuate each segment. The cables (⌀0.38 mm nylon wire) are threaded through their respective ⌀0.8 mm channels that are isometrically distributed in a radius of 2.1 mm. A 2-mm-diameter main channel is reserved at the axial center. The robot base is mounted on a linear slide for a feed motion along the axial direction. Our design features a proximal segment of 60 mm and a distal segment of 70 mm in length, with a unified diameter of 6.2 mm to imitate a stylet or an endoscope and ease of fabrication and assembly. The soft bodies were made from Agilus30 photopolymer using a PolyJet 3D printer (J826 Prime, Stratasys).

Each cable is winded on a spool mounted on a bearing on the fixture base. The spools can be rotated by the couplers that are connected to the flexible shafts actuated by the respective DC motor (1000:1 gear ratio, 6V) with an encoder for angular feedback. The DC motors are PID-controlled by a low-cost self-assembled motion controller equipped with three L293D units and a general microcontroller. The linear slide is driven by a stepper motor drive. With some step-down transformers, all electronic components are well-fitted in a portable acrylic box with a standard power cable (220V AC) and a USB port for communication with the PC. Such spatial-mechatronic-decoupled design reduces the footprint size, increases the system's portability, and facilitates the free space posing of the robot base.

IV. Simulation: SOFA-Based Robot Modeling

A. Virtual Environment

We configured the simulation in SOFA v22.06.99. To align with the physical scene where the robot tip would be facing toward the ground, we set the virtual gravity to be $[0, 0, 9.81]^\top$

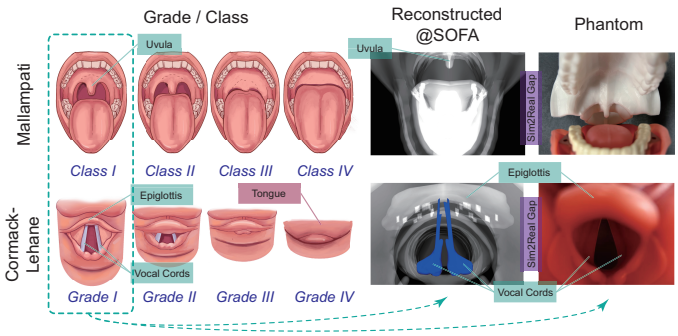

Fig. 4. This work is predicated on encountering a uvula with a Class-I visibility in the Mallampati system and the vocal cords with a Grade-I visibility in the Cormack–Lehane system.

m/s^2. As the baseline, the current work is predicated on ideal anatomic scenes with clear visibility of the airway anatomic structures, with the uvula of class-I visibility in Mallampati score [32] and vocal cords of grade-I classification in Cormack–Lehane system [33]. The assumption is further depicted in Fig. 4. It should be noted that unless extensive data with various anatomic conditions are used, the reconstructed and phantom environments we present may not fully reflect the complexity and variety of real cases. The sim-to-real discrepancy at the current visibility class/grade can be further reduced by using Fourier domain adaptation [34] and style transfer [35] to even achieve pixel-grade cross-domain (SOFA/phantom) feature segmentation [36]. A modified oropharyngeal-tracheal 3D phantom [37] was directly imported into the scene in `obj` format. To reduce the expensive finite element computation, we trimmed some insignificant entities from the phantom, such as teeth and miscellaneous muscles, leaving the phantom with 26,706 triangular surfaces, as shown in Figs. 5(a) and 5(b).

B. Robot Modeling

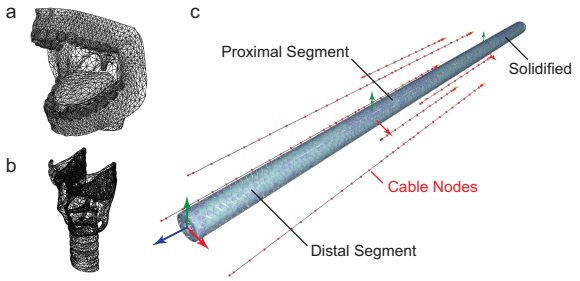

Fig. 5. Entities modeling in SOFA. (a) Meshed modified oral cavity, and (b) meshed pharynx and trachea. Adapted from [37], [38] under CC BY-NC-SA license. (c) Cables' geometric constraints (exploded view) of the meshed two-segment cable-driven robot.

In SOFA, robot modeling depends on the meshed solid model of the soft bodies and their geometric constraints, including cable distribution, actuation regulation, and partial solidification. To do that, the soft robot was first sketched in FreeCAD (a free and open-source software under the LGPL-2.0-or-later license) and exported to `brep` or `step` format. Then, we imported the model into Gmsh (a free software under the general public license) for the meshing and exported it to both `vtk` and `stl` format. The `vtk` was used to add the finite element model in SOFA, and the `stl` was used to define the visual model and collision model. After some trial-and-errors with the consideration of computational cost and rationality, we tetrahedrally meshed the soft segments into voxels with 4985 vertices, excluding any cable channels in the mesh.

Based on the prototype fact, we resembled the physical properties in SOFA with the Young's modulus $E = 0.8$ MPa, Poisson's ratio $\nu = 0.45$, and the total mass $m_{soft} = 7$ g. The cable actuation mechanism was geometrically constrained in the python script, which can be expressed as

$$L_{i,j,k} = \begin{bmatrix} (-1)^j \cdot r_c \cdot \sin\big((j-1)\,\beta\big) \\ -r_c \cdot \cos\big((j-1)\,\beta\big) \\ (k-1)\,d \end{bmatrix} \tag{1}$$

where $i = 1$ and $i = 2$ represents the proximal and distal segment, respectively; $j = \{1, 2, 3\}$ denotes the indexed cable; and $k = \{1, 2, ..., N_i\}$ indicates the kth node along the soft body. Note that N_i varies from different segments, and $N_2 > N_1$. For the constants, $r_c = 2.2$ mm is the radius for cable distribution, $\beta = \frac{2\pi}{3}$ is the angular offset between the neighboring cables of the same segment, and $d = 5$ mm denotes the sampling distance along the soft body. As for the rigid shaft, it has to be solidified as a rigid part. The working range of the prismatic joint along the feeding axis was set to be $[0, 80)$ mm. Figure 5 gives an intuitive illustration of our robot modeling in SOFA.

C. Collision Avoidance Motion Planning

With all the preparations ready, we then defined a relative position between the soft robot's base and the environment. For the sake of convenience and valid computation, we located the robot base frame $\{B\}$ at $P_{target} = [0, -55, 160]^\top$ mm with respect to the target site as a basic status (also refer to Figs. 1, 9). The relative frames relationship can also be found in the weak registration in the real-world deployment which will be discussed in Sec. VI.

The built-in QPInverseProblemSolver was used for the collision avoidance motion planning. The cost function is a quadratic function that minimizes the actuation and the distance between robot meshes (a function of the actuation) and the environment. The solver implements the QP problem with linear complementarity constraints (QPCC) [20] based on the qpOASES library to inversely compute the corrected FEM-based robot model in response to the actuators, actuator constraints, and surroundings. Different primitives, including point, line, and triangle, were utilized in the narrow phase intersect detection. A local minimum distance proximity method was used to evaluate the anticipation of contact with an alarming distance of 2 mm and a contact distance of 0.5 mm. The pseudo-code in algorithm 1 depicts the collision-free navigation deployment workflow in SOFA. Fig. 6 demonstrates two resultant examples in collision avoidance and evaluation with the said proximity method. As the robot moves, the objective values from the QP formulation converge.

Since we would be interested in the eye-in-hand vision, a camera frame $\{C\}$ was additionally attached at the robot's tip

Algorithm 1 ESTABLISH ENVIRONMENT & ROBOT MODEL IN SOFA TO COMPUTE COLLISION-FREE MOTION FOR NAVIGATION

procedure ROOTNODE()
  requiredPlugins ▷ SOFA SoftRobots Plugin
  defaultVisualManagerLoop & freeMotionAnimationLoop
  visualStyle & gravity ▷ Robot appearance; $G = 9.8$ m/s^2
  collisionPipeline: alarmDistance $= 2$, contactDistance $= 0.5$
  QPInverseProblemSolver (epsilon $=$ 1e-1) ▷ Compute Inverse
  simulationNode()
    - solversForDeformation: OdeSolver, linerSolver, SparseLDLSover, GenericConstraintCorrection ▷ Compute soft object deformation
    - softRobot: FEM, visual, collisionModel ▷ `vtk, stl`
      - rigidify() ▷ Rigid shaft
      - deformablePart: cableActuators ▷ Cable nodes using Eq. (1)
      - rigidPart: slidingActuator ▷ Linear slide
    - mechanicalMatrixMapper (rigidAndDeformableCoupling)
  phantomModel(visual, collisionModel) ▷ `obj`
  define frames: target & end-effector
  recordedCamera: orientation & position from myAnimation
  animate(myAnimation)
  return

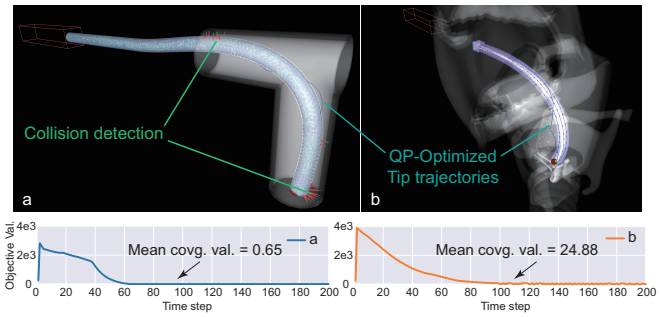

Fig. 6. Simulation snapshots of the collision avoidance motion planning of our soft robot in different SOFA scenes using the built-in QP solver and the local minimum distance-based proximity method, interacting with (a) a 90-deg pipe, and (b) the phantom in Sec. IV-A. The respective objective values are given. The default initial objective values are 250.

to provide an endoscopic view. For simplicity, we ignored that offset between the camera and the tip frame and coincided with them, i.e., $\{C\} \approx \{T\}$. Given that there are no available tools to acquire the endoscopic view, we defined a fixed frame on the plane perpendicular to all cable ends – based on trigonometry – as the camera frame. The coordinates of three cable ends can be indicated w.r.t. the robot base as $p_j = L_{2,j,N_2} \in \mathbb{R}^{3\times 1}$ where $j = \{1, 2, 3\}$. Then, the plane formulated by those coordinates can be calculated by

$$\begin{bmatrix}\alpha & \beta & \gamma\end{bmatrix}^\top = (p_1 - p_2) \times (p_3 - p_2)\,. \tag{2}$$

Thereby, the orientation of camera frame can be computed by

$$C = \left(\begin{bmatrix} \arccos \sqrt{\frac{\alpha^2+\gamma^2}{\alpha^2+\beta^2+\gamma^2}} \\ \arctan\left(\frac{\alpha}{\gamma}\right) \\ 0 \end{bmatrix} \right)_{\otimes} \cdot R_x(\pi) \cdot R_z(\pi)\,, \tag{3}$$

where $(\cdot)_{\otimes}$ denotes the operations that convert an Euler angle to a rotation matrix, and $R_x(\theta)$, $R_z(\theta)$ represent the rotation matrices of θ on the subscripted axis. The rotation matrix C was then converted to a quaternion for use. The frame origin was located at $\frac{1}{3}(p_1 + p_2 + p_3)^\top$. With the specific definition of viewport coordinate and focal length, the eye-in-hand view can be acquired from the QtViewer using OpenGL.

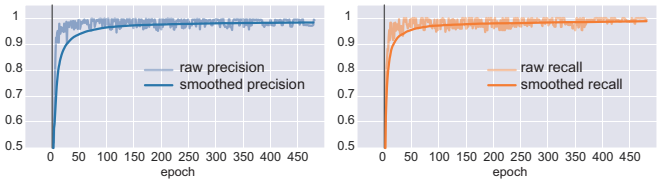

Fig. 7. Performance metrics of the YOLOv5s for anatomical recognition. Early stopping was triggered at epoch 481 as no improvement was observed in the last 100 epochs. Best results observed at epoch 380.

This method would grant us theoretically unlimited virtual endoscopic image data of anatomical features/organs, as long as we can build in SOFA, without concerning privacy issues.

V. LEARNING FROM THE VIRTUAL VISION

In this work, the learnings can be divided into two parts and will be introduced in this section. Sec. V-A describes the use of primarily SOFA-generated images (i.e., eye-in-hand viewport) with a small number of phantom pictures blended for anatomical feature recognition. While Sec. V-B depicts the recurrent learning between the SOFA-generated joint space motion – subjected to the QP-constraints for collision avoidance – and the resulting labels of recognized features in the virtual environment.

A. Recognizing Anatomical Features using YOLO

We employed YOLO (You Only Look Once) [39], a real-time object detection algorithm, for the online anatomical feature recognition task. Due to the limitation in available virtual 3D models, the SOFA environment is over-animated, which fails to satisfy the feature recognition task in the real world. To bridge the gap between simulation and reality in this regard, we blended the simulated endoscopic dataset with some pictures of the phantoms. The dataset size is given in Table I. The images in the dataset were labeled using bounding boxes with corresponding feature tags. The dataset was arbitrarily divided into training (80%), validation (10%), and test set (10%). Among the four released models (https://github.com/ultralytics/yolov5), namely the 5s, 5m, 5l, and 5x, we opted for the most lightweight YOLOv5s model. The model was set to be trained for 800 epochs with a batch size of 4, and the early stopping (patience at 100) was triggered at the 481st epoch, meaning that the best results were observed at epoch 380. The network performance is given by Fig. 7 and Table II, showing that it can classify the three classes with a high precision of 0.989, 1, 0.989, for uvula, epiglottis, and glottis, respectively. As an indicator metric for object detection, Table II also explicitly provides the mean average precision (mAP) for intersection over union (IoU) greater than 0.5 and from 0.5 to 0.95. Figures 8 and 9 demonstrate the effectiveness of the trained feature recognition model in both simulation and reality scenes, where the recognized features could be parameterized into the respective category (see the Z-axis of Fig. 11) with coordinated bounding boxes in real-time. The generalization of feature recognition can be improved further by taking into account anatomic appearances with varying visibility grades introduced in Sec. IV-A.

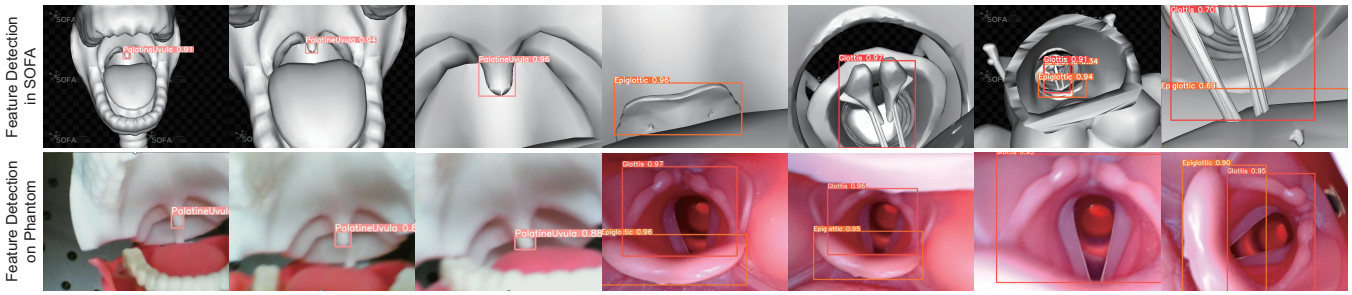

Fig. 8. The trained PyTorch model (YOLOv5s) can effectively recognize and track single or multiple anatomical features (uvula, epiglottis, glottis) in simulation and real environments. The above examples were fed to the model with confidences $= 0.7$.

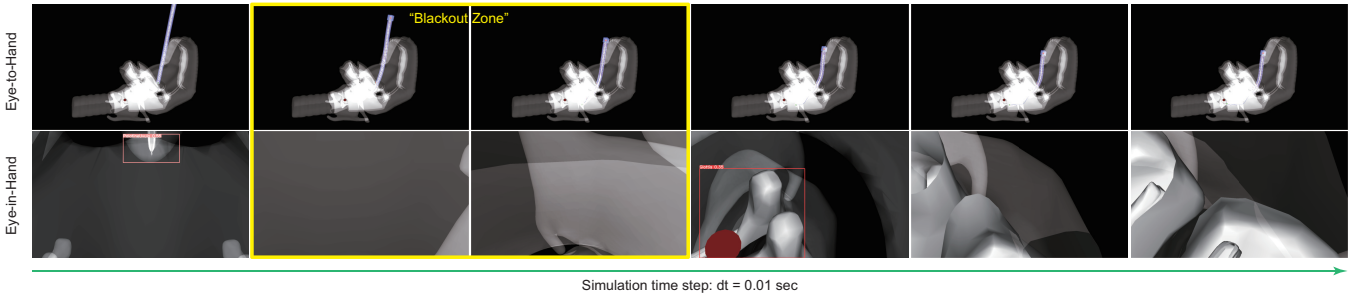

Fig. 9. Snapshots of one of the SOFA simulation groups: Based on the built-in QPInverseProblemSolver, a set of feasible actuators' solutions that avoid physical body collision can be derived and fed back into the simulator for visualization. By associating a camera frame $\{C\}$ to the robot's tip, a simulated endoscopic vision (i.e., eye-in-hand) can be obtained for further feature learning. The view encounters a "blackout" period without recognizing any key features.

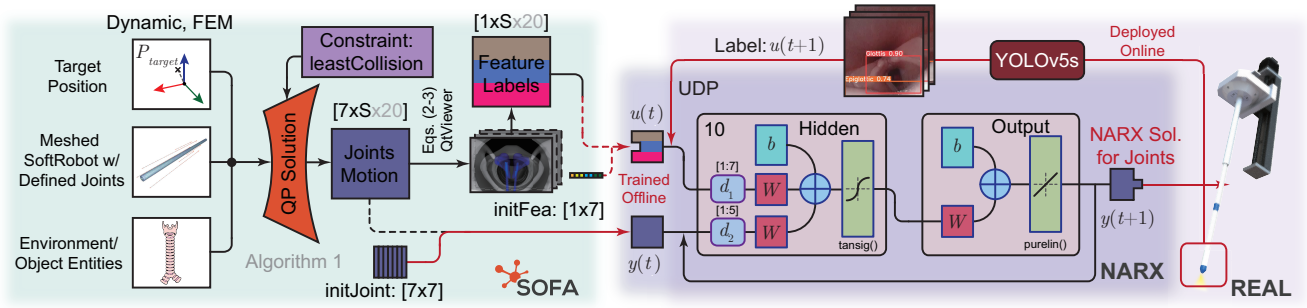

Fig. 10. System architecture of the proposed sim-to-real transfer learning method: The system uses the simulation data from SOFA for the NARX network training. While in a real environment, the system codes the recognized features from its real eye-in-hand vision (labeled as $\boldsymbol{u(t+1)}$) and current joint space motion $\boldsymbol{y(t)}$ to forecast the next move $\boldsymbol{y(t+1)}$. Here the **S** refers to the length of simulation timestep.

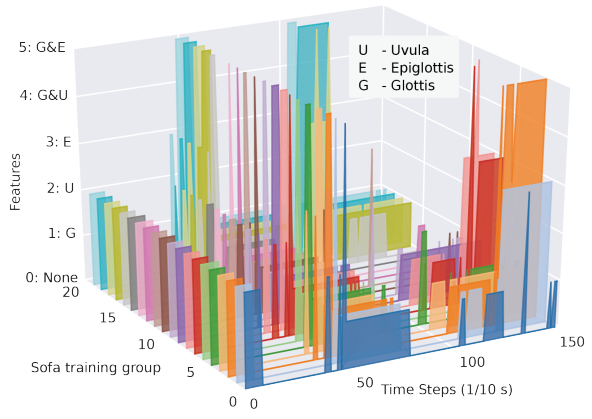

Fig. 11. With variable targets, 20 groups of feature sequences obtained by the SOFA's eye-in-hand vision were utilized for the NARX training.

TABLE I
SIZE OF THE BLENDED DATASET FOR FEATURE RECOGNITION (UNIT: FRAME)

	Uvula	Epiglottis	Glottis
SOFA's Virtual Images	601	395	507
Colored Real Images	119	69	82

B. Eye-Hand Learning using NARX

Since the virtual endoscopic images and the robot actuation ("eye-hand") are temporal dependents, the nonlinear autoregressive exogenous (NARX) model [40] was used for the learning. NARX is a class of discrete-time nonlinear models that are often utilized as an open-loop or closed-loop form multistep predictor in times series modeling. The advantage of using the NARX is that the whole operation can be involved by a single model. Such a model can be algebraically represented

TABLE II
MODEL SUMMARY OF YOLOV5S-BASED FEATURE RECOGNITION

Class	Test Images /Instances	Precision	Recall	mAP @0.5	mAP @0.5:0.95
All	140 / 181	0.993	0.993	0.995	0.921
Uvula	140 / 72	0.989	1	0.995	0.877
Epiglottis	140 / 51	1	0.980	0.994	0.928
Glottis	140 / 58	0.989	1	0.995	0.957

by [28]:

$$y(t+1) = f\left[y(t), y(t-1), \cdots, y(t-n_y+1), u(t), u(t-1), \cdots, u(t-n_u+1)\right] \quad (4)$$

where $y(t)$ and $u(t)$ are, respectively, the output and the input sequence of the network at the discrete time step of t. Meanwhile, n_y and n_u are the delays in output and input, respectively, subject to $n_u \geq n_y \geq 1$. The dependant output value at the next time step $y(t+1)$ is regressed on its previous output and previous independent exogenous input.

Since we cannot provide a perfect driver sequence $y(t)$ in prior to the NARX network for prediction, we need to train the network in a closed-loop way, i.e., using the newly predicted driver sequence as part of the input, then combining it with the visually recognized labels for the next prediction. Besides, we improved the training process regarding the model generalization and overfitting avoidance by using the early stopping method with automated regularization under the Bayesian framework [41]. The model training was implemented using the Neural Network Toolbox of Matlab. In the NARX network, d_1 and d_2 denote non-negative input delays and output (feedback) delays, respectively. These hyperparameters must be tuned based on the specific problem and data characteristics, and no reference values exist. Here, we empirically initialized the input delays as $d_1 = [1 : n_u]$ and the output delays as $d_2 = [1 : n_y]$, where $n_u = 7$ and $n_y = 5$. For the initialization of network training, we found that the NARX network would produce more stable initial predictions if the input delay were replenished with some small non-zero values at the first seven timesteps, which is in response to the input delay. In our work, we supplemented the SOFA-generated joint motion with arbitrary small values as

$$y_{init} = 10^{-4}\begin{bmatrix} | & | & \cdots & | \\ 1 & 2 & \cdots & 7 \\ | & | & \cdots & | \end{bmatrix} \quad (5)$$

for each training. It has been tested that using other small values for initialization would not cause a significant difference. Whereas, the initial amended labels can be $u_{init} \in \mathbb{R}^{1\times 7}$ as long as it does not interpret any executable features.

After training a total of 300 NARX networks, we kept the network with the most satisfactory performance. The selection was made by feeding each network with 10 sets of new label data from YOLO and driven data from SOFA exclusive from the original dataset for training, validation, and testing, and obtaining the mean squared error performance for comparison. To diversify the simulation data, we added a cohort of offsets of $[\pm 3, \pm 3, \pm 5]^\top \in \mathbb{Z}$ in millimeter on the target position P_{target} in three axes, resulting in 20 groups of simulation paths in SOFA. Using offset targets for the training could contribute to weak robot-patient registration in a real-world deployment. The offsets were also selected to fulfill the consequence that each path would correspond to a unique endoscopic view and QP-solved actuation. Figure 11 illustrates the sequences of feature(s) captured by the eye-in-hand vision in SOFA. The variable target positions affect the virtual visual perception in terms of timing, duration, and recognized features, enriching the simulation data. Such variation mimics the slight individual difference in physiological appearance among people, which would benefit the model's adaption to new oropharyngeal environments.

The architecture diagram in Fig. 10 summarizes our proposed method. The robot modeling and the feature recognition were implemented in Python and YOLOv5 (PyTorch framework), respectively, while the eye-hand learning using the NARX network was performed in Matlab R2020a. The networks were trained on an NVIDIA GeForce RTX 3060 GPU. Since multiple platforms were involved, we employed a user datagram protocol (UDP) socket that allows the Python program to stream the real-time recognition to where the Matlab host on the actuator side could receive it.

VI. EXPERIMENT

A. Experiment Setup

A simplified robotic TI scene was set for the experiments. A CMOS image sensor (OV6946, OmniVision, CA, USA) was used to provide eye-in-hand vision. The 1.8-mm-diameter LED-equipped image sensor can capture 400 × 400 resolution video stream at a 30 fps frame rate. To show the generalization of the proposed method, in the experiment, we used a commercially-available clinical oropharyngeal phantom, which was different from the simulated scene that produced the eye-hand dataset for NARX network training. The experiment setup is shown in Fig. 12. The linear slide that holds the manipulator was vertically installed with the robot tip pointing toward the ground and fixed on an adjustable holder. A permanent magnetic-based tracking system was used to obtain the 3D position of a magnet attached to the robot tip. A tiny NdFeB magnet was used to avoid excessive payload. Based on the magnet's size, the valid measurement range of the tracking system is about 100 mm above the array.

Before configuring the sim-to-real method, we tested the robotic system with open-loop control. As shown in Fig. 12, two tip paths were imported to the SOFA to derive the inverse solution in joint space. The paths (in mm) for the circle and ∞-shape are, in $t = 0 : \pi/500 : 2\pi$ time steps, $x_{ref,o} = 16{\cdot}\sin(t)$, $y_{ref,o} = 16 \cdot \cos(t)$, and $x_{ref,\infty} = 22 \cdot \mathrm{sgn}(\cos(t)) \circ (\cos(t) \circ \cos(t))$, $y_{ref,\infty} = 22 \cdot \mathrm{sgn}(\cos(t)) \circ \sin(t) \circ (\cos(t) \circ \cos(t))$, respectively, with a height of 52 mm above the magnetometer array. The $\circ$ operator denotes the element-wise (Hadamard) product. Due to the unit problem, the SOFA-generated inputs necessitate an overall amplification to fit the prototype setup, such as spool sizes and the minor transmission losses of using flexible shafts. The measured results show that our setup can reproduce the desired path with an average spatial positioning error below 2 mm in open-loop mode, which is adequate for using the latter sim-to-real validation with closed-loop control.

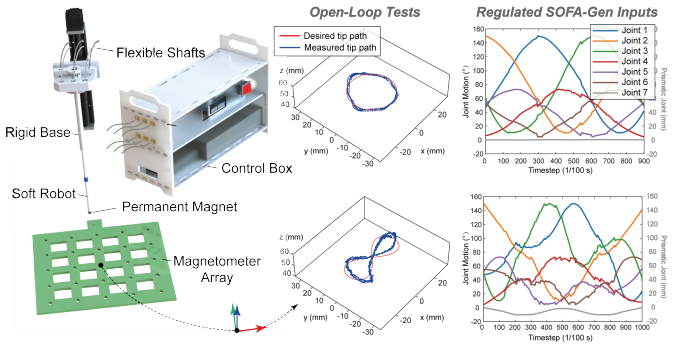

Fig. 12. Experiment setup and the open-loop control tests. The robot joint space motions are computed by the QP solver of the SOFA framework. The tip positions are captured by the magnetometer array.

B. NARX Performance

We performed a prior experiment in the simulation to evaluate the closed-loop NARX network performance in our joint space motion forecasting task. Initially, three new target positions proximate to the upper glottis were randomly selected in SOFA. Notably, the selected target position was intentionally excluded from the training, validation, and testing datasets to prevent biases. After running the simulations, we obtained the SOFA-generated joint space motion with a time-series feature sequence observed by the endoscopic vision in the virtual scene. After that, we tried to feed the recorded feature to the trained NARX in sequence – imitating a real-time feature sequence that the actual camera vision would attain – and examined the alignment between the forecasted and SOFA-generated joint space motion. The result is given in Fig. 13. It can be seen that the endoscopic vision would observe the features differently in terms of time and the ROI of features. Such variant observed features and the closed-loop mechanism would result in NARX-forecasted joint motions that are deviated from the reference joint motions computed by the QP solver of SOFA. However, the deviations are insignificant for the overall robot motion. As shown in Fig. 13, the cable joint of the proximal segment, joints 1–3, are nearly merged with only minor differentiation. The resultant task motion showed that such joint motion would stiffen the proximal motion to antagonistically resist the passive bending motion caused by the distal segment, which conforms to the literature [42], [43] and simulation. Table III demonstrates the R-squared coefficient of determination of the motion of each joint for the above experiments. The R-squared coefficient is given by

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}},$$

where SS_{res} and SS_{tot} denote the residual sum of squares and the total sum of squares, respectively. It measures how well the NARX network forecast can fit the SOFA-generated outcomes. Based on the above three experiments, the table shows that the average R^2 coefficient for the cable joints is 0.963, with the lowest performance shown in joint 3 (one of the cable joints for the proximal segment). In contrast, joint 7 (the prismatic joint) has an average R^2 coefficient of 0.997. The prior results validate the fidelity between what the trained

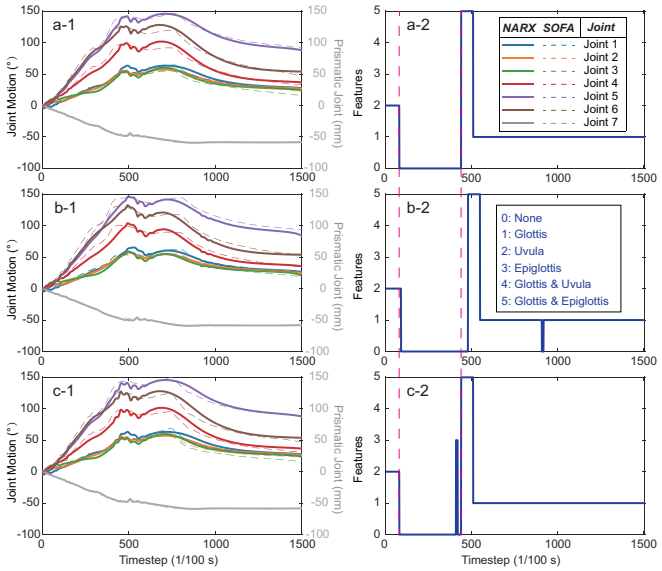

Fig. 13. Closed-loop NARX-forecasted joint space motions based on only the virtual eye-in-hand vision, compared with the ideal SOFA-generated joint space motion. Three examples are given in (a), (b), and (c), with variant target positions of $[\mathbf{0}, \mathbf{-55}, \mathbf{155}]^\top$, $[\mathbf{3}, \mathbf{-55}, \mathbf{155}]^\top$, and $[\mathbf{0}, \mathbf{-58}, \mathbf{155}]^\top$ mm, respectively. The resultant observed features are shown in the right column.

TABLE III
R-SQUARED FITNESS FOR FIG. 13

	Joint	1	2	3	4	5	6	7 (P)
R^2	Exp. (a)	0.981	0.978	0.909	0.965	0.992	0.981	0.998
	Exp. (b)	0.972	0.975	0.926	0.962	0.990	0.978	0.998
	Exp. (c)	0.962	0.959	0.894	0.960	0.985	0.972	0.997

NARX network produces and the computation from the SOFA. It supports the closed-loop sim-to-real implementation in the following Sec. VI-C.

Moreover, a comparison was made between the performance of the NARX network and the Long Short-Term Memory (LSTM) network, which is frequently employed for learning time-series sequential data. The trained networks were assigned a new target position of $[0, -55, 155]^\top$ mm for joint motion predictions, and the outcomes were subsequently contrasted with the joint motion computed by the QP solver in SOFA. Fig. 14 presents a side-by-side comparison of the network predictions. The R^2 coefficients for the LSTM and open-loop NARX networks are 0.983 and 0.997, respectively, and the training time for these open-loop models take respectively about 120 seconds and 10 seconds with an Intel i9 12th-gen CPU using Matlab. Nevertheless, when fitting the predicted and referenced outputs based on such a single trial, the NARX network exhibited a mean square error (MSE) that was 15.7% lower than LSTM. These results suggest that the NARX network outperforms LSTM in terms of performance with less training time. Furthermore, the NARX network has a simpler architecture, requires less computation, and exhibits better generalization and robustness to input changes than LSTM especially when only small amounts of training data are available [44]. Here, training a usable closed-loop NARX network for 3,000 epochs takes about 12 minutes on the same PC configuration. And thanks to the relatively short training

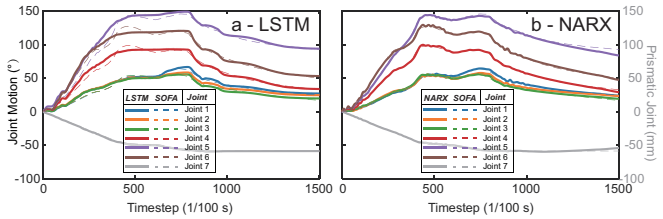

Fig. 14. Given a target position of $[\mathbf{0}, -\mathbf{55}, \mathbf{155}]^\top$ mm, the above provides joint space motions predicted by (a) an LSTM; and (b) an open-loop NARX. The former demonstrates an $\boldsymbol{R^2}$ coefficient of 0.983 and an MSE of 0.4419, while the latter shows an $\boldsymbol{R^2}$ coefficient of 0.997 and an MSE of 0.3820, respectively.

time, we could explicitly test the training effect, adjust the hyper-parameters promptly, and select the optimal model in due course.

C. Validating the Sim-to-Real Transfer on Phantom

An overview of our sim-to-real-driven deployment is shown in Fig. 15. A video is also available in the supplementary material. Using a phantom different from the simulation, the soft robot's tip was placed about 15 mm above the uvula to resemble the initial scene of the SOFA's as much as possible. When operated, the robot would navigate to the upper glottis location based on only its real-time eye-in-hand vision. The vision underwent the YOLO algorithm for feature detection, and the recognized feature sequences were decoded into executable joint space motion by the trained-NARX network. Fig. 16 demonstrates the selected video sequence from the eye-in-hand vision of two experiments with slightly different initial settings of the relative positions between the robot and phantom. Their corresponding recognized features are also given in Fig. 16. Even with different initial placements, it can be observed that the robot can efficaciously navigate to the desired location without significant collision with the environment, which is unstructured in any of our simulations. The measured tip paths during the navigation are available in Fig. 17, which also indicate the variability of the network's decision depending on what the eye-in-hand vision receives in real time. As we have planned in the SOFA scene, the soft robotic endoscope/stylet manipulation should

1) avoid colliding with the uvula at the beginning;
2) perform the first major bending at the oropharynx; and
3) distinguish esophagus and glottis, then align to the latter.

To evaluate the above criteria, (1) can be visually examined, (2) can be verified by post-evaluation of tip position measurement, and (3) can be visually evaluated by the real-world endoscopic view.

The experiment results showed that the robot could conduct the given navigation task automatically based on what it had learned from the simulations. Due to the different initial settings, the robot "saw" different feature sequences, resulting in diverse NARX-forecasted joint space motions as shown in Fig. 18 – both of them are capable of driving the robot to fulfill the task. The stiffening effect on the proximal soft segment due to the antagonistic actuation was also realized on the prototype as planned in the physical simulation. As designed in the simulation, the proposed method allows a relative malpositioning between the robot and the phantom. The experimental observation suggested an error tolerance of $[0 \pm 5, -55 \pm 5, 160 \pm 5]^\top$ mm of P_{target} in $\{B\}$, which surpasses to the simulation. The success rate of sim-to-real transfer has usually been one of the indicators to evaluate deploying performance [45]. Table IV shows the success rate of the sim-to-real transfer in reaching the upper glottis in our phantom experiments. Here, a success reach was judged by whether it could provide a clear endoscopic view showing the vocal cords or not. While the simulation can always obtain a viable view at the end, a high success rate of 17 out of 20 consecutive trials were found capable of delivering the camera into the spots in real-world phantom experiments. In our observation, failed trials were primarily accused of overexposure due to the intense LED light (can tell from Exp. 2 in Fig. 16), which interferes with YOLO-detection when acquiring valid features for closed-loop feedback.

TABLE IV
SIM & REAL PERFORMANCE – SUCCESS RATE OF REACHING THE UPPER GLOTTIS

	SOFA Sim.	Real
Phantom Used	Modified 3D-modeled Phantom [37]	Commercially-available Phantom
Success Rate	20/20	17/20

D. Discussion

The major novelty of our visual-dependent sim-to-real method for soft robots can be highlighted as the following when compared with:

1) General Eye-in-Hand Visual Servoings: In general, visual servoing (VS) requires detected features in the loop at all times. But oftentimes, the endoscope sees nothing or invalid frames (this has been verified in simulation and real phantom), which is inadequate for valid closed-loop feedback. In contrast, our sim-to-real method can refer to the "memories" that the robot learns from the virtual world, allowing its navigation without relying on a continuous eye of sight of features. If we use an eye-in-hand VS method in our task, it may require many feature labels along the navigation to determine the next step, whereas ours only requires a few, significantly reducing the time-consuming training in feature recognition. Furthermore, compared to the traditional VS method, a sim-to-real method reduces the onsite calibration of the visual system's extrinsic and intrinsic matrices [46], initial Jacobian estimation between the joint and task space [47], and position-configuration measurement [48]. In addition, general eye-in-hand VS may not be able to control the whole-body motion of a multisegment soft robot, which is essentially required for the navigation task in a tortuous environment, while our method is capable of whole-body motion control. Due to privacy and ethical concerns, the medical dataset used to deploy visual servoing is typically inaccessible. However, simulation scenes are more accessible and customizable. Computer graphics experts might contribute to more realistic virtual scenes for future sim-to-real deployment. Therefore, as supplementary to

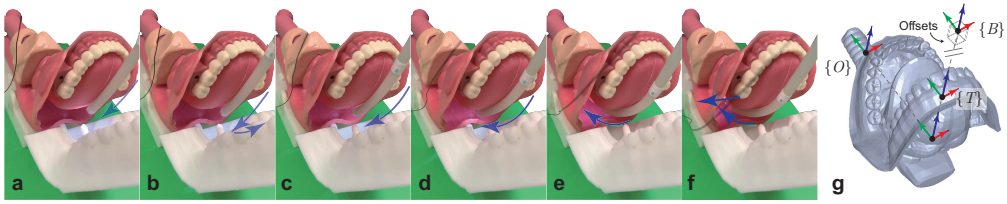

Fig. 15. (a–f) Snapshots of the robot motion generated by the proposed sim-to-real method. The joint space motions were automatically computed based on the endoscopic vision and the closed-loop NARX algorithm. (g) Weak robot-patient registration.

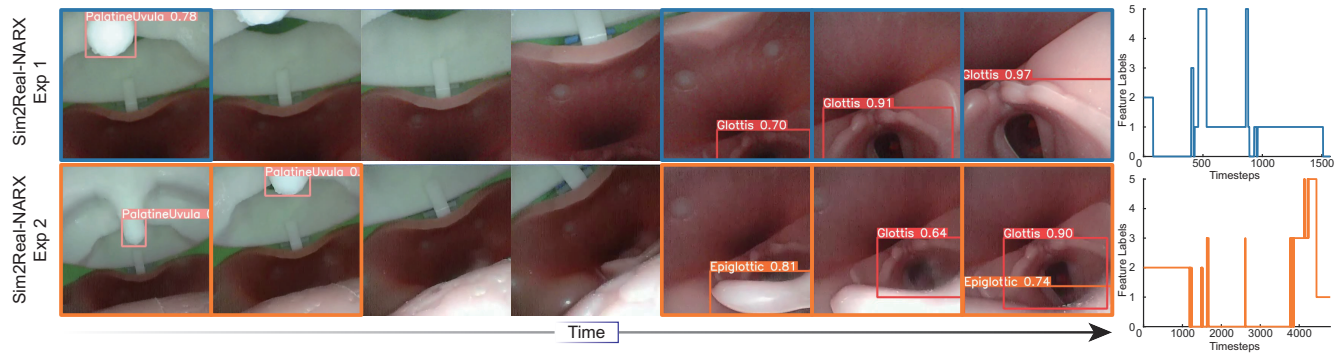

Fig. 16. Snapshots from the eye-in-hand vision of two experiments with different initial settings of the relative positions between the robot and phantom and their corresponding recognized features (0: None; 1: Glottis; 2: Uvula; 3: Epiglottis; 4: Glottis & Uvula; 5: Glottis & Epiglottis).

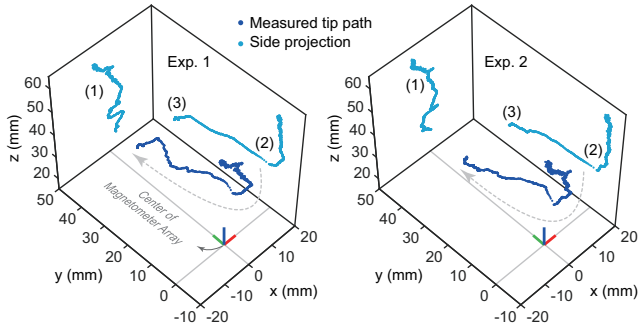

Fig. 17. The measured NARX-forecasted tip motion for (a) Experiment 1 and (b) Experiment 2 of Fig. 16. Numbered Tags: (1) Avoid colliding with the uvula. (2) First major bending at the oropharynx. (3) Second major bending, aligning to the glottis instead of the esophagus.

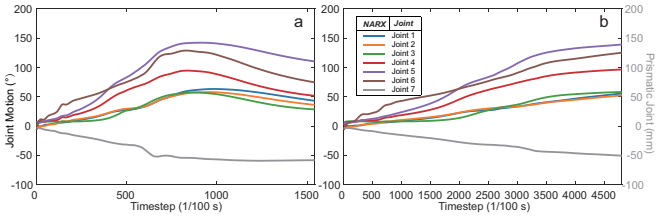

Fig. 18. The recorded NARX-forecasted joint space motion for (a) Experiment 1 and (b) Experiment 2 of Fig. 16.

the VS methods, a vision-based sim-to-real method like ours will be meaningful for the developers and roboticists.

2) Model-Based Simulation Frameworks: Our soft robot simulation is based on the open-source SOFA framework, instead of model-based simulators. Due to its physics engine, the framework is friendly to deformable entities of soft materials with multiple collision models and collision detection methods, providing a rich source of simulation data for the sim-to-real transfer. The use of embedded eye-in-hand vision in the virtual environment, which is not available in general model-based simulators, improves transfer fidelity as well.

3) Deep Learning Frameworks: A light-weighted network like NARX is more suitable for our desired application. There are only a few anatomical features in the human oropharynx structure, which can be handily covered by the permutation of explainable feature labels. However, a deep learning network requires more expensive computational overhead as it would also account for the voided vision that further challenges the sim-to-real discrepancy. Also, the explainable feature labels with pathologies/defectives can be added to further enrich the simulation dataset in a separate recognition training process.

4) Possible Extension to Real-World Anatomy: We have also explored the possibility of further extending part of this work to real anatomical application. One of the critical parts will be reducing the discrepancy between simulation and reality regarding the YOLO-based feature recognition. Following a similar strategy, we blended the datasets with some real medical images from open access sources [49], [50] to train a model applicable in simulation, phantom, and anatomical environment. Detailed configuration of the datasets is given in Table V, indicating a comparative mAP to the model that was trained using only SOFA's and phantom images. To further verify the newly introduced model, we input new video clips (i.e., excluded from the learning process) into the model after some necessary trimming (4:21–4:32 and 4:57–5:51) for the recognition test. As shown in Fig. 19, the model was able to recognize the intended features, even though the real anatomical images only account for 7.61% of the dataset. The preliminary results reveal the possibility of applying the proposed sim-to-real transfer strategy in future real-world trials.

TABLE V
COMPARISON OF DATASET THAT BLENDS WITH PHANTOM IMAGES ONLY & WITH ADDITIONAL MEDICAL IMAGES FROM

Dataset (#)	# of Blends (%)	mAP @0.5	mAP @0.5:0.95
SOFA (1194) Phantom (203)	203/1397 (14.53%)	0.995	0.925
SOFA (1194) Phantom (203) Medical (115)	115/1512 (7.61%)	0.995	0.910

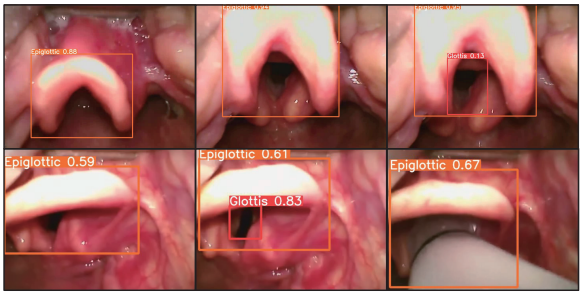

Fig. 19. Recognizing real-world anatomical features from open accessible clips [51] using a model trained with mostly (i.e., 92.39%) simulation and phantom data.

5) Limitation: We acknowledge that there are also some limits to our current work. For instance, the method lacks perceptual/control mechanisms to deal with possible collisions in different real-world environments. Such drawbacks can be further improved by employing additional haptic sensors and including them in the simulations to diversify the virtual sensing (other than only vision) in future works. Moreover, the proposed method can be further completed by introducing variations on virtual scenes to reduce the discrepancy among patients.

VII. CONCLUSION

This work proposes a SOFA-based sim-to-real method for soft robotic navigation that learns from the virtual eye-in-hand vision using the NARX network. Motivated by the soft robotic endoscope/stylet manipulation procedure before the transoral TI, this work firstly presents a two-segment 3D-printed cable-driven soft robotic system featuring a miniature manipulator, soft material, and mechatronic-decoupled design. Based on the prototype and open-source 3D phantom models, a virtual environment that resembled the soft robot navigation during stylet manipulation was reconstructed in SOFA. The SOFA's built-in QP solver was used to compute the minimal collision motions in both task space and joint space. Meanwhile, eye-in-hand visions in the virtual world were obtained. The YOLOv5 algorithm was configured to recognize the observed anatomical features, namely, the uvula, glottis, and epiglottis, with a high precision of over 98.9% in virtual and phantom environments.

Then, using only the simulation data, a closed-loop NARX network was trained to associate the time series anatomical features sequence with the SOFA-generated joint space motion. After that, the trained network was employed in the real-world soft robotic system. Equipped with eye-in-hand vision, the soft robot with the NARX network could compute the joint space motion in real-time based on what it observes, despite the diverse environment and robot-phantom setting, and autonomously navigate to the desired spot with minimal collision to the environment. The experiment results show that the soft robot can efficaciously navigate through the unstructured phantom to the desired spot near the upper glottis with minimal collision motion according to what it has learned from SOFA. The average R-squared coefficient between the closed-loop NARX-forecasted and SOFA-referenced robot's cable and prismatic joint space motion are 0.963 and 0.997, respectively. The eye-in-hand visions demonstrate good alignment between the robot tip and the glottis.

ACKNOWLEDGMENT

The authors would like to thank Dr. Hugo Talbot and Dr. Eulalie Coevoet, who promptly replied to our inquiries on GitHub when we had trouble using SOFA and SOFTROBOTS plugin.

REFERENCES

[1] M. Cianchetti, C. Laschi, A. Menciassi, and P. Dario, "Biomedical applications of soft robotics," *Nat. Rev. Mat.*, vol. 3, no. 6, pp. 143–153, 2018.
[2] C. Li, X. Gu, X. Xiao, C. M. Lim, and H. Ren, "A robotic system with multichannel flexible parallel manipulators for single port access surgery," *IEEE Trans. Ind. Inform.*, vol. 15, no. 3, pp. 1678–1687, 2018.
[3] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1261–1280, 2015.
[4] 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.
[5] D. C. Rucker and R. J. Webster III, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Trans. Robot.*, vol. 27, no. 6, pp. 1033–1044, 2011.
[6] C. Li, X. Gu, X. Xiao, C. M. Lim, and H. Ren, "Flexible robot with variable stiffness in transoral surgery," *IEEE/ASME Trans. Mechatron.*, vol. 25, no. 1, pp. 1–10, 2019.
[7] H. Wang *et al.*, "Visual servoing of soft robot manipulator in constrained environments with an adaptive controller," *IEEE/ASME Trans. Mechatron.*, vol. 22, no. 1, pp. 41–50, 2016.
[8] T. G. Thuruthel, B. Shih, C. Laschi, and M. T. Tolley, "Soft robot perception using embedded soft sensors and recurrent neural networks," *Sci. Robot.*, vol. 4, no. 26, p. eaav1488, 2019.
[9] C. Duriez, "Control of elastic soft robots based on real-time finite element method," in *Proc. IEEE Int. Conf. Robot. Autom, (ICRA)*, 2013, pp. 3982–3987.
[10] S. Song, Z. Li, M. Q.-H. Meng, H. Yu, and H. Ren, "Real-time shape estimation for wire-driven flexible robots with multiple bending sections based on quadratic bézier curves," *IEEE Sens. J.*, vol. 15, no. 11, pp. 6326–6334, 2015.
[11] H. Wang, R. Zhang, W. Chen, X. Liang, and R. Pfeifer, "Shape detection algorithm for soft manipulator based on fiber bragg gratings," *IEEE/ASME Trans. Mechatron.*, vol. 21, no. 6, pp. 2977–2982, 2016.
[12] Y. Chen *et al.*, "Modal-based kinematics and contact detection of soft robots," *Soft Robot.*, vol. 8, no. 3, pp. 298–309, 2021.
[13] J. Lai, B. Lu, Q. Zhao, and H. K. Chu, "Constrained motion planning of a cable-driven soft robot with compressible curvature modeling," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4813–4820, 2022.
[14] S. H. Sadati *et al.*, "Tmtdyn: A matlab package for modeling and control of hybrid rigid–continuum robots based on discretized lumped systems and reduced-order models," *Int. J. Robot. Res.*, vol. 40, no. 1, pp. 296–347, 2021.
[15] X. Huang, X. Zhu, and G. Gu, "Kinematic modeling and characterization of soft parallel robots," *IEEE Trans. Robot.*, 2022.
[16] W. Huang, X. Huang, C. Majidi, and M. K. Jawed, "Dynamic simulation of articulated soft robots," *Nat. Commun.*, vol. 11, no. 1, pp. 1–9, 2020.
[17] D. S. Shah *et al.*, "A soft robot that adapts to environments through shape change," *Nat. Mach. Intell*, vol. 3, no. 1, pp. 51–59, 2021.

[18] J. Hiller and H. Lipson, "Dynamic simulation of soft multimaterial 3d-printed objects," *Soft Robots.*, vol. 1, no. 1, pp. 88–101, 2014.

[19] H. Choi *et al.*, "On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward," *Proc. Natl. Acad. Sci.*, vol. 118, no. 1, p. e1907856118, 2021.

[20] E. Coevoet, A. Escande, and C. Duriez, "Optimization-based inverse model of soft robots with contact handling," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1413–1419, 2017.

[21] E. Coevoet *et al.*, "Software toolkit for modeling, simulation, and control of soft robots," *Adv. Robot.*, vol. 31, no. 22, pp. 1208–1224, 2017.

[22] T. Zhang, K. Zhang, J. Lin, W.-Y. G. Louie, and H. Huang, "Sim2real learning of obstacle avoidance for robotic manipulators in uncertain environments," *IEEE Robot. Autom. Lett.*, vol. 7, no. 1, pp. 65–72, 2021.

[23] M. Kaspar, J. D. M. Osorio, and J. Bock, "Sim2real transfer for reinforcement learning without dynamics randomization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2020, pp. 4383–4388.

[24] J. Z. Zhang *et al.*, "Sim2real for soft robotic fish via differentiable simulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2022, pp. 12 598–12 605.

[25] U. Yoo, H. Zhao, A. Altamirano, W. Yuan, and C. Feng, "Toward zero-shot sim-to-real transfer learning for pneumatic soft robot 3d proprioceptive sensing," *arXiv preprint arXiv:2303.04307*, 2023.

[26] S. Kriegman *et al.*, "Scalable sim-to-real transfer of soft robot designs," in *Proc. 3rd IEEE Int. Conf. Soft Robot. (RoboSoft)*, 2020, pp. 359–366.

[27] J. D. Greer, L. H. Blumenschein, R. Alterovitz, E. W. Hawkes, and A. M. Okamura, "Robust navigation of a soft growing robot by exploiting contact with the environment," *Int. J. Robot. Res.*, vol. 39, no. 14, pp. 1724–1738, 2020.

[28] J. M. P. Menezes Jr and G. A. Barreto, "Long-term time series prediction with the narx network: An empirical evaluation," *Neurocomputing*, vol. 71, no. 16-18, pp. 3335–3343, 2008.

[29] J. Ong *et al.*, "A new video intubating device: Trachway® intubating stylet," *Anaesthesia*, vol. 64, no. 10, pp. 1145–1145, 2009.

[30] Q. Boehler *et al.*, "Realiti: A robotic endoscope automated via laryngeal imaging for tracheal intubation," *IEEE Trans. Med. Robot. Bionics.*, vol. 2, no. 2, pp. 157–164, 2020.

[31] P. W. Furlow and D. J. Mathisen, "Surgical anatomy of the trachea," *Ann. Cardiothorac. Surg.*, vol. 7, no. 2, p. 255, 2018.

[32] S. R. Mallampati *et al.*, "A clinical sign to predict difficult tracheal intubation; a prospective study," *Can. Anaesth. Soc. J.*, vol. 32, pp. 429–434, 1985.

[33] R. Cormack and J. Lehane, "Difficult tracheal intubation in obstetrics," *Anaesthesia*, vol. 39, no. 11, pp. 1105–1111, 1984.

[34] Y. Yang and S. Soatto, "Fda: Fourier domain adaptation for semantic segmentation," in *Proc. IEEE/CVF Conf. Compt. Vis. Pattern Recognit.*, 2020, pp. 4085–4095.

[35] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. IEEE Int. Conf. Compt. Vis. (ICCV)*, 2017, pp. 1501–1510.

[36] G. Wang, T.-A. Ren, J. Lai, L. Bai, and H. Ren, "Domain adaptive sim-to-real segmentation of oropharyngeal organs," *Med. Biol. Eng. Comput.*, 2023.

[37] University of Dundee, School of Medicine, "Pharynx and Floor of Mouth," https://skfb.ly/6QXqr, accessed: 2022-08-01.

[38] Cubebrush, "Photorealistic human mouth," http://cbr.sh/fixcgf, accessed: 2022-08-01.

[39] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788.

[40] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, "Learning long-term dependencies in narx recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 7, no. 6, pp. 1329–1338, 1996.

[41] D. J. MacKay, "Bayesian interpolation," *Neural Comput.*, vol. 4, no. 3, pp. 415–447, 1992.

[42] J. Lai, B. Lu, and H. K. Chu, "Variable-stiffness control of a dual-segment soft robot using depth vision," *IEEE/ASME Trans. on Mechatron.*, vol. 27, no. 2, pp. 1034–1045, 2021.

[43] M. Manti, V. Cacucciolo, and M. Cianchetti, "Stiffening in soft robotics: A review of the state of the art," *IEEE Robot. Autom. Mag.*, vol. 23, no. 3, pp. 93–106, 2016.

[44] A. Wunsch, T. Liesch, and S. Broda, "Groundwater level forecasting with artificial neural networks: a comparison of long short-term memory (lstm), convolutional neural networks (cnns), and non-linear autoregressive networks with exogenous input (narx)," *Hydrol. Earth Syst. Sci.*, vol. 25, no. 3, pp. 1671–1687, 2021.

[45] A. Byravan *et al.*, "Nerf2real: Sim2real transfer of vision-guided bipedal motion skills using neural radiance fields," *arXiv preprint arXiv:2210.04932*, 2022.

[46] F. Xu, H. Wang, W. Chen, and Y. Miao, "Visual servoing of a cable-driven soft robot manipulator with shape feature," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4281–4288, 2021.

[47] J. Lai, K. Huang, B. Lu, and H. K. Chu, "Toward vision-based adaptive configuring of a bidirectional two-segment soft continuum manipulator," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron. (AIM)*, 2020, pp. 934–939.

[48] K. Wu *et al.*, "Safety-enhanced model-free visual servoing for continuum tubular robots through singularity avoidance in confined environments," *IEEE Access*, vol. 7, pp. 21 539–21 558, 2019.

[49] Cook Children's Health Care System, "Laryngoscopy Education Video," https://youtu.be/hVQ4EZW63qQ, accessed: 2023-05-18.

[50] OpenAirway, "Adult AirTraq intubation with reinforced endotracheal tube," https://youtu.be/xI2epkL69wc, accessed: 2023-05-18.

[51] InterAnest, "Endotracheal Intubation," https://youtu.be/8AOB2PtHfVM, accessed: 2023-05-18.