

Department of Mechanical, Materials and Manufacturing Engineering

Faculty of Engineering

Design and Control of a Continuum Robot with Multiple Embedded Sensors

By

Yihua FANG

Bachelor of Mechanical Engineering

Thesis submitted to

the University of Nottingham

for the degree of Doctor of Philosophy

May 2023

Abstract

Continuum robots are a type of hyper-redundant manipulator, which is generally structured with multiple small diameter sections and can access and operate in narrow & hazardous environments. This research project aims to explore the shape-sensing technology and feasible control strategy for a multiple-section continuum robot system, to solve the control challenge introduced by the hyper-redundancy (i.e., the number of actuators is less than the number of DOFs) and coupled motion (i.e., for the robot with more than one section).

Through this work, a hybrid sensing system is developed and implemented in a prototype continuum robot, with a two-level closed-loop controller developed and a vision-aided led-by-the-tip navigation strategy proposed based on it. This sensing solution consists of a dual displacement sensor system and a self-carried visual tracking system. The former works to measure the cable displacement error at the actuation pack and predict the cable elongation in the arm structure. The latter equips the robot with several camera-marker arrays along the arm and a head camera at the distal tip, providing section-by-section tip tracking and visual perception for the robot controller. Integrated with the two-level closed-loop controller, the whole system is designed to minimize control error in both joint-space and task space. Further, an adaptive led-by-the-tip navigation strategy is proposed to navigate the robot arm into confined space, with a head camera observing the working environments. Under this work frame, the robot can snake its way through narrow channels and perform environmental interactive tasks such as obstacle avoidance and object detection.

All the works were demonstrated on a two-section cable-driven continuum robot with a rotary actuation system design, mounted in a linear rail to provide feed-in motion. Each section is 120 mm long and allows 2-Dof motion in 3-D space. It was shown that the measurement error of the proposed self-carried visual tracking system is smaller than 2 mm, with a mean error of 0.67 mm within the workspace of a 120 mm single section. The distal-tip control accuracy of the 2-section prototype robot can achieve 1.72 mm in the world coordinate (in the X-Y plane), which reduces the tip control error by 73.5% compared with

the conventional PCC (piecewise constant curvature) model-based controller, for both smooth and sharp movements. In open-space navigation tasks, along a 240 mm guiding path, the robot achieved an average trajectory error of 3.5 mm with a tip-following deviation of 2.5 mm. Then, the robot is navigated into a mock-up with boundaries and obstacles to demonstrate its capability for online obstacle detection and motion planning. These experiments validate the feasibility and reliability of the sensing system and controller architecture proposed in this study.

Publications

The research in this thesis has contributed in part or full to the following publications:

(1) **Fang, Y**., Dong, X., Mohammad, A. and Axinte, D., 2023. Design and Control of a Multiple-Section Continuum Robot With a Hybrid Sensing System. IEEE/ASME Transactions on Mechatronics.

(2) Mohammad, A., Russo, M., **Fang, Y.,** Dong, X., Axinte, D. and Kell, J., 2021. An efficient follow-the-leader strategy for continuum robot navigation and coiling. IEEE Robotics and Automation Letters, 6(4), pp.7493-7500.

(3) **Fang, Y.,** Hill, S., Ba, W. and Dong, X., 2022, July. A Novel Method to Solve the Inverse Kinematics for Continuum Robots Accelerated by PID Theory. In 2022 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER) (pp. 840-844). IEEE.

(4) **Fang, Y.,** Bishop, C., Ba, W., Díez, J.B., Mohammad, A. and Dong, X., 2020. A universal stiffening sleeve designed for all types of continuum robot systems. In Towards Autonomous Robotic Systems: 21st Annual Conference, TAROS 2020, Nottingham, UK, September 16, 2020, Proceedings 21 (pp. 15-24). Springer International Publishing.

Acknowledgments

I would like to sincerely thank my parents, Mr. Fang and Ms. Hu, for giving birth to me and supporting me throughout my life, your prayer for me was what sustained me this far.

I would like to express my sincere gratitude to my academic supervisors Dr. Xin Dong, Prof. Dragos Axinte, and Dr. Abd Mohammad, thanks for your guidance and support carried me through all the stages of my research work.

This Ph.D. project is funded by the University of Nottingham and Rolls-Royce University Technology Centre for the initial three years. Acknowledgment of the financial support to allow me to finish my study.

Last but not least, my sincere thanks also go to all the staff and colleagues in the UTC, for the time we worked together and for all the happiness we shared.

Table of Contents

ABSTRACT I

PUBLICATIONS III

ACKNOWLEDGMENTS IV

TABLE OF CONTENTS V

LIST OF FIGURES X

LIST OF TABLES XIX

NOMENCLATURE XX

CHAPTER 1 INTRODUCTION 1

1.1 Research background 1

1.2 Problem definition 2

1.3 The prototype robot 3

1.3.1 The twin-pivot robot arm 3

1.3.2 The rotary driving system 5

1.4 Objectives of this study 6

1.5 Structure of the thesis 7

1.6 Highlights and contributions 8

1.6.1 Comprehensives sensing solution for a continuum robot system 8

1.6.2 A novel two-layer control strategy based on the new sensing solution 9

1.6.3 Novel vision-aided navigation approach for continuum robot 10

CHAPTER 2 LITERATURE REVIEW 11

2.1 Continuum robot design 11

2.1.1 Tendon-driven continuum robots 11

2.1.2 Pneumatic continuum robots 13

2.1.3 Other types of continuum robots 14

2.2 Continuum robot modeling 17

2.2.1 Kinematics model 17

2.2.2 Dynamics model of continuum robots 21

2.3 Sensing solutions for continuum robot (shape sensing & tip tracking) 22

2.3.1 Contact sensing solutions for shape sensing 23

2.3.2 Non-contact sensing solutions 27

2.3.3 Other sensing methods 29

2.3.4 Summary of general sensing solutions 30

2.3.5 Embedded vision applications on continuum robot systems 31

2.3.6 Embedded vision applications on other robotic systems 33

2.3.7 Summary of embedded vision applications 35

2.4 Controller designs for continuum robots 36

2.4.1 Model-based controllers 36

2.4.2 Model-less controllers 39

2.5 Navigation for continuum robots in the task space 41

2.5.1 Navigation technologies for conventional robot systems. 42

2.5.2 Navigation technologies for continuum robots 44

2.6 Opportunities for future research 45

CHAPTER 3 BASIC MODELING, FORMULATION, AND CONTROL CHALLENGES FOR THE PROTOTYPE ROBOT 47

3.1 Basic concepts in continuum robot control 47

3.2 Kinematics between configuration space to task-space 48

3.2.1 The PCC forward kinematics 48

3.2.2 The modified PCC-based single-variable inverse kinematics 51

3.3 Kinematics from configuration space to joint space 53

3.4 Modeling and control challenges for the prototype robot system 56

3.5 Chapter summary 60

CHAPTER 4 A HYBRID SENSING SYSTEM INTEGRATED CLOSED-LOOP CONTROLLER DESIGNED FOR A MULTIPLE-SECTION CONTINUUM ROBOT 61

4.1 Concept of the hybrid sensing system 63

4.2 The two-level closed-loop controller based on the hybrid sensing system 65

4.2.1 The Joint-space cable displacement controller 66

4.2.2 The work-space tip position controller based on the paired observe-reference camera & marker arrays 71

4.3 Chapter summary 81

CHAPTER 5 EXPERIMENTS AND VALIDATION FOR THE HYBRID SENSING BASED CLOSED-LOOP CONTROLLER 82

5.1 The continuum robot system setup for experiments 82

5.2 Experiment results 84

5.2.1 Measurement accuracy test for the self-carried visual tracking system 84

5.2.2 Validation for the dual displacement sensor system based two-stage cable displacement control approach 86

5.2.3 Validation for the two-level closed-loop controller with the feedback from the hybrid sensing system 88

5.3 Chapter summary 91

CHAPTER 6 VISION-BASED CLOSED-LOOP NAVIGATION FOR CONTINUUM ROBOTS IN STRUCTURED ENVIRONMENTS 93

6.1 Overview of the navigation strategy 93

6.1.1 The robot system setup for navigation 94

6.1.2 The control strategy of robot navigation 94

6.2 Stage 1 – Path pre-planning: the led-by-the-tip strategy 96

6.2.1 Discretized guiding path 96

6.2.2 The led-by-the-tip strategy 98

6.3 Stage 2 - Vision aided navigation with on-line obstacle avoidance & path re-planning 102

6.3.1 Obstacle detection 102

6.3.2 On-line path replanning 104

6.4 Stage 3 - Target detection and measurement 105

6.4.1 The working frame of target measurement 105

6.4.2 Method of pseudo-triangulation based on a single (head) camera 106

6.5 The remote controller developed for navigation tasks 107

6.6 Experiment and validation for the navigation strategy 110

6.6.1 An overview 110

6.6.2 Closed-loop navigation in an open space under the led-by-the-tip strategy 111

6.6.3 Navigating into a mock-up of restricted spaces 121

6.6.4 Object detection and measurement 124

6.7 Chapter summary 129

CHAPTER 7 CONCLUSION AND FURTHER WORK 130

7.1 Academic finding and engineering validation 130

7.1.1 Novel hybrid sensing solution developed to evaluate and feedback system errors of continuum robots 130

7.1.2 The closed-loop controller developed for a multiple-section continuum robot (a model-less control strategy) 132

7.1.3 An adaptive navigation strategy developed to conduct the robot enter more complex working scenarios 133

7.2 Further work 134

7.2.1 Sensing solutions for continuum robots 134

7.2.2 Robot controller design 135

7.2.3 Further development on the vision-based sensing system and environment interactivity 136

REFERENCE LIST 138

List of Figures

Figure 1-1 Concept drawing of a slender continuum robot, with structure design of robot arm and actuation system [9]. 1

Figure 1-2 The 2-section cable-driven prototype robot. 3

Figure 1-3 The pivot-joint robot design. (a) structure of a disk; (b) structure of a segment; (c) a bended section consists of several segments driven by actuation cables. 4

Figure 1-4 The rotary actuation design. 5

Figure 2-1 Examples of tendon-driven continuum robots. (a) A snake-like robot actuated by push-pull rods [22]. (b) A continuum robot driven by actuation cables [24, 25]. 12

Figure 2-2 Examples of the actuation systems of tendon-driven continuum robots. (a) The linear actuation system employs linear motors for driving the tendons [17]. (b) The rotary actuation unit utilizes rotary motors and spool systems to drive the tendons [15]. 13

Figure 2-3 Examples of air muscles composed of elastic tubes and constraints [29].. 13

Figure 2-4 Operating spaces of a continuum manipulator and their definitions [19]. . 17

Figure 2-5 A constant curvature geometry [88], where k is the curvature of the curve, $\emptyset$ is the rotation if the curve and s is the arc length. 18

Figure 2-6 2-D H-D geometry a constant curvature continuum robot (single section) [18]..... 18

Figure 2-7 3D D-H geometry of a constant-curvature continuum robot (single section). (a) The virtual 5-joint robot [88]. (b) Geometry representation of the robot [18]. 19

Figure 2-8 An example of typical neural network architecture [89]. 20

Figure 2-9 Neural network-based model learning of CBHA [90, 91]. (a) The robot configuration. (b) Inverse Neural Network learning..... 20

Figure 2-10 Examples of stretchable conductive material applications for building continuum robots. (a)A McKibben actuator, an example of electro-conductive yarn-based shape sensing [99]. (b) The Smart Braid, an example of inductance-based shape sensing [100]. (c) A soft pneumatic actuator (SPA), an example of liquid metal-based shape sensing [101]..... 23

Figure 2-11 Examples of displacement sensors for measuring tendon displacement. (a) An example of applying potentiometers on a cable-driven continuum robot to control tendon displacement [36]. (b) An example of using passive cable displacements to estimate robot shape [108]. (c) An example of using a linear encoder to control cable displacement [13, 16]. (d) An example of a pneumatic robot equipped with potentiometer cables to estimate the position of the end-effector [58]..... 25

Figure 2-12 Examples of optical sensors. (a) A macro bend sensing system made of optical fiber [109]. (b) A sensor sleeve design containing tactile sensors and optical fiber [110]. (c) A deflection sensor composed of LED and light sensor [111]. (d)The inner structure of FBG sensors. (e) An example of FBG shape sensor application [112]. ... 26

Figure 2-13 Examples of EM tracker and vision & imaging systems for measuring the configurations of continuum robots. (a) Application of EM tracker on a multiple-section continuum robot [39]. (b) Application of VICON system (infrared cameras) in tip-tracking [13]. (c) Shape sensing by stereo camera system [122]. (d) An example of using the x-ray to reconstruct the configuration of a medical robot [117]. 28

Figure 2-14 Examples of novel tip-position sensing methods for continuum robots that are not commonly used. (a) Shape estimation using permanent magnets and magnetic sensors [123]. (b) Triangulation system based on wire-potentiometers [58]..... 30

Figure 2-15 A summary of sensing technologies for continuum robots..... 30

Figure 2-16 Applications of embedded visions on continuum robots for imaging and sensing. (a) A slender continuum robot designed with two monocle cameras embedded for in-situ inspection of air engines [9]. (b) A fetoscope equipped with tip cameras designed for placental scanning [125]. (c) A continuum robot arm and gripper for

harvesting cherry tomatoes [126]. (d) Endoscopic images used to control the flexible instruments [127]. (e) An eye-in-hand application of a tip-attached camera on a continuum robot [128]. (f) Application of a tip-attached camera [129]. 32

Figure 2-17 Basic conceptual framework of visual servoing (vision-based control) of robotic systems [135]. (a) The eye-in-hand configuration. (b) The eye-to-hand configuration. (c) A basic dynamic position-based look-and-move servoing architecture. 34

Figure 2-18 Combined configuration of a visual servoing. (a) Eye-in-hand / eye-to-hand cooperation for visual servoing [133]. (b) Eye-to-hand approach on eye-in-hand configuration [125]. 35

Figure 2-19 Architects of two configuration space controllers. (a) A sliding-mode configuration controller designed for Octarm (extensible pneumatic robot) for planner operation [53]. (b) A configuration controller designed for a cable-driven continuum robot [39]. 37

Figure 2-20 A comparison between the joint-space and task-space controllers [139]. (a) The joint-space control algorithm. (b) The task-space control algorithm. (c) The prototype robot. (d) The tip-tracking performance of the two controllers compared with open-loop control. 38

Figure 2-21 Examples of model-less & model-free control of continuum robots. (a) RNN-based compliant force control of a cable-driven continuum robot[140]. (b) RNN-based tension prediction [140]. (c) Adaptive neural network control for CBHA: the controller architecture [58]. (d) Adaptive neural network control for CBHA: the path tracking experiment [58]. (e) The control diagram of the LWPR learning-based control [98]. 40

Figure 3-1 The operating spaces of the prototype continuum robot with corresponding control variables. 47

Figure 3-2 The PCC (piecewise constant curvature) assumption of a single-section continuum robot. 48

Figure 3-3 An example of a 3-section continuum robot, whose shape curves with continuous tangent vectors. 50

Figure 3-4 Inverse kinematics for a single-section robot (based on bisection method). 51

Figure 3-5 Sketch for a multiple-section problem. 52

Figure 3-6 Workflow chart of the multi-section inverse kinematics. 53

Figure 3-7 Cable hole distribution through the disks of the prototype robot, rh is the radius of the cable hole pattern. 54

Figure 3-8 Graphical representation of actuation cables passing through the disks. ... 54

Figure 3-9 Illustration for the equivalent radius of a specific cable hole when the bending angle is φ. 55

Figure 3-10 Cases of unideal winding of cables on the spools. 57

Figure 3-11 Error analysis for the prototype continuum robot. 59

Figure 4-1 An overview of the hybrid sensing system and the cooperative two-level closed-loop controller. 61

Figure 4-2 The prototype of the robot equipped with the sensing system. (a) The robot architecture. (b) The reference coordinates are defined in each section. 62

Figure 4-3 The concept of a hybrid sensing system. (a) The sensing system composed of a dual displacement sensor system and a self-carried visual tracking system. (b) A 3-D sketch of the self-carried visual tracking system. 63

Figure 4-4 Control diagram for the two-level closed-loop controller supported by the hybrid sensing system. 65

Figure 4-5 Integration of the linear encoder system: the measurement unit, the encoder set, and the whole robot system. 67

Figure 4-6 Basic definitions in joint-space control. (a) Illustration of a single-section robot with a self-carried camera. (b) The dual displacement sensor system (Lm: measurements recorded by rotary encoders; Ll: the measurements recorded by linear encoders). 68

Figure 4-7 The two-stage cable-displacement control approach based on linear and rotary encoders measurement. 69

Figure 4-8 A single-section continuum robot with a paired observe-reference camera & marker array..... 74

Figure 4-9 The calibration and 3-D restoring process for the visual system. 74

Figure 4-10 Setup for geometry calibration between robot coordinate and visual system. 75

Figure 4-11 Explanation for the indirect measurement based tip-adjustment strategy. (a) A front view of the robot work-space (projection view). (b) Marker offset vectors in ¼ work-space (the blue dots: robot tip position; the black circles: visual marker position). 77

Figure 5-1 System overview. a) the two-section continuum robot with the hybrid sensing system. b) a measurement unit of the linear-encoder system, the actuation cables are clamped by the lockable frame where the encoder chip attached; c) System communication. 83

Figure 5-2 Experiment Setup: the off-the-shelf tracking (VICON) system provides a reference tip-tracking to evaluate the tip-positioning performance in the experiment. 84

Figure 5-3 Tip-tracking performance. Tip trajectories recorded by the off-the-shelf tracker (blue) and camera & marker array (red) are compared for benchmarking the accuracy of the vision system integrated into the continuum arm. 85

Figure 5-4 Validation for the linear encoder system. (a) workflow diagram for the experiment evaluating the rotary actuation system; (b) cable distribution for the tested section; (c) comparison between theoretical cable displacement (solid line) vs actual cable displacement (dot line). 87

Figure 5-5 The tip-positioning accuracy comparison: open-loop control performance comparison of the single-section robot under dual-displacement sensor feedbacked control (right) and rotary encoder control (left)..... 88

Figure 5-6 The tip-positioning accuracy comparison: error vectors comparison, under rotary encoder control (left) and dual-displacement sensor system control (right). 88

Figure 5-7 Tip-positioning performance in reference (local) coordinate: open-loop & closed-loop control accuracy comparison for both sections. 89

Figure 5-8 Control accuracy of the distal tip under closed-loop control supported by the hybrid sensing system in the world coordinate. (a) A 3-D view of the desired path of the desired tip. (b) A project view of the tip-trajectory in the x-y plane (black: desired path; red: the actual path of the distal tip; blue: the actual path of the base-section-tip)..... 90

Figure 6-1 The robot system setup for the navigation task. The 2-section snake arm is mounted on a linear rail to provide linear motion. 94

Figure 6-2 The work flowchart for the navigation process, which is divided into 2 stages: path pre-planning and navigating into the working environment. In the first stage, a guiding list is generated according to desired tip position; for the second stage, the robot goes into the working environment while functions like obstacle avoidance are achieved. 95

Figure 6-3 Discretization for a single-section continuum robot. 97

Figure 6-4 The discretized m-step guiding path $[\boldsymbol{xtip}, \boldsymbol{ytip}, \boldsymbol{ztip}]\boldsymbol{m}$ for a 2-section prototype robot. 98

Figure 6-5 Illustration for the starting pose of the robot (black points), desired guiding path (white circle) and desired robot pose for a specific step. 99

Figure 6-6 Sketch of robot pose fitting. The predicted robot is discretized into several subsections and represented by a group of sampling points with the same resolution as the planned path. 100

Figure 6-7 Simulation of the navigation process based on the guiding list $[\theta g, \varphi g, Zg]m$, the solid line in black represents the pre-planned path for the robot, the blue arc illustrates the tip section, and the red arc presents the base section, the pink straight lines represent the camera & marker arrays at the base of each section. (a) step 1: initial pose

of the snake. (b) step 10. (c) step 20. (d) step 30. (e) step 40. (f) step 49: the final pose of the robot at the end of the navigation, fully inserted into the desired path..... 101

Figure 6-8 Working principle for obstacle detection. 103

Figure 6-9 Method of path re-planning. When an obstacle occurs in the pre-planned path and is detected by the snake robot at the current location, the controller will re-direct the planned path according to user input to avoid the obstacle. 104

Figure 6-10 Communication channels between MATLAB and LabVIEW for the task of object measurement. 106

Figure 6-11 Flowchart of the pseudo-triangulation method based on the head camera. (a) An animation. (b) The working principle of the triangulation process. 107

Figure 6-12 An overview of the controller App: the data communication, the working scenario, and a working demo. 108

Figure 6-13 The user interfaces (UI) of the entry page and navigation bar of the controller. 108

Figure 6-14 UIs of the remote controller: (a) the 'Controller' page. (b) the 'Navigation' page. List of main buttons: ① control stick for bending angle; ② control stick for direction angle; ③ emergency stop; ④ section selection bar; ⑤ velocity limitation bar; ⑥ numerical control instructions indicator; ⑦ navigation coordinate selecting window; ⑧ for forward/ backward button. 109

Figure 6-15 Experiment setup for navigation in a constructed environment. (a) the robot arm setup, the body camera arrays are attached at the section tips and the head camera is installed at the distal tip. (b) the mock-up for the demo, the red-dash line represents the pre-planned path to navigate the robot into the target area. 111

Figure 6-16 Graphic explanation of experiment results. (a) The instant VICON record of the robot base (red dot), the tip of a body section (Tip1, blue dot), and the distal tip (Tip 2, green dot) vs desired trajectories (pre-planned path). (b) Illustration of the path error (instant distal tip position vs discretized guiding path points). (c) Illustration of

the trajectory error (instant distal tip position vs guiding path curve). (d) Illustration of the tip-following error (trajectory difference between the distal tip and the following tip(s) of body section(s)). 112

Figure 6-17 An example of a navigation process: the instant position recording of the robot tracked by VICON (red: robot base; blue: section tip; green: distal tip) against the desired robot trajectory (pre-planned guiding path, the solid black line). 114

Figure 6-18 Robot trajectories recorded by VICON in the first-quadrant navigation experiment. (a) Under open-loop control mode. (b) Under closed-loop control mode. 115

Figure 6-19 Error comparison (graphic) between open-loop navigation and closed-loop navigation in the first quadrant. (a) The path errors. (b) The trajectory errors. (c) The tip-following errors. 117

Figure 6-20 Robot trajectories recorded by VICON in the second-quadrant navigation experiment. (a) Under open-loop control mode. (b) Under closed-loop control mode. 119

Figure 6-21 Error comparison (graphic) between open-loop control and closed-loop control (an experiment in the second coordinate). (a) The path errors. (b) The trajectory errors. (c) The tip-following errors. 121

Figure 6-22 setups of the mock-up. (a) ① the mock-up; ② the continuum robot; ③ the linear rail. (b) the inner structure of the experiment mock-up. 122

Figure 6-23 Performance of the robot under open-loop navigation when passing through the second barrier. 123

Figure 6-24 Performance of the robot under closed-loop navigation when passing through the second barrier. 123

Figure 6-25 Performance of the robot under closed-loop navigation with the implementation of obstacle avoidance algorithm when passing through the second barrier. 124

Figure 6-26 The experiment setup for object detection and measurement. (a) The robot system, the mock-up, and the detected pattern. (b) The detected pattern. 124

Figure 6-27 Procedure for shape detection: Step 1: read the raw image. Step 2: convert to a binary image. Step 3: noise reduction. Step 4: shape recognition. 125

Figure 6-28 (a) Image matching. (b) the data measured in this experiment. 126

Figure 6-29 Graphic presentations of the results. (a) the measurement results for each image pair. (b) the measurement error for each image pair. (c) the measurement results sorted according to measurement error. 128

List of Tables

Table 1 General types of continuum robots..... 14

Table 2 Robot Navigation Technologies 44

Table 3 Cable hole distribution angles for both sections. 56

Table 4: Measurement accuracy of the self-carried visual tracking system compared with the Off-the-shelf Position Tracker (The Vicon System)..... 86

Table 5 Tip-positioning error for the two-section robot in the local coordinate..... 91

Table 6 Tip-positioning error for the two-section robot in the world coordinate 91

Table 7 System capabilities in navigation work. 110

Table 8 Error comparison (quantitively) between open-loop and closed-loop control for the experiment in the first coordinate. 116

Table 9 Error comparison (quantitively) between open-loop and closed-loop control for the experiment in the second coordinate..... 120

Table 10 Distance measurement results from multiple image pairs (part). 127

Table 11 Summary of the measurement results for object measurement. 127

Nomenclature

Symbol	Description	Units
l	Actuation cable displacement.	mm
ω_m	The rotation angle of the actuation motors.	rad
p	The pitch of the winding of the actuation spool (in the motor-spool driving system).	mm
d	Diameter of the actuation spool.	mm
(θ, φ)	Joint parameters: bending angle and direction angle of a robot arm.	rad
r_{sec}	The radius of the arc of a bent section.	mm
s	Section length of the robot arm.	mm
P_x	Position vector.	
R_x	Rotation matrix.	
${}^{a}_{b}T$	Transformation matrix from a to b.	
$[x, y, z]$	Cartesian coordinate system.	
$Rot_K(\emptyset)$	Rotation operator which performs a rotation about the axis $\widehat{K}$ of a reference coordinate by $\emptyset$ degrees.	
r_h	The radius of the cable hole pattern relative to the center of the disc.	mm
r_e	The equivalent radius of a specific cable hole relative to the reference (bending) plane.	mm
ε	The angle of a specific cable hole relative to the reference coordinate of a section.	rad
L_*	Actuation cable length or displacement.	mm
j	Number of disks in a section.	
α_e	Optimal elevation angle of the self-carried cameras.	rad
d_p	Optimal spacing robot arm axial and self-carried cameras.	mm
A_*	Camera field of view (FOV).	mm
T_*	Range of motion (ROM) of the tip-attached marker.	mm
f_m	The offset of the marker relative to the robot tip.	mm

l_c	Length of a probe camera.	mm
$\alpha_{v/h}$	The view angle of a probe camera (vertical or horizontal).	rad
f_c	The offset of the probe camera relative to the robot axis.	mm
S_m	The side length of the marker	mm
E_x	The extrinsic parameters of cameras.	
$\overrightarrow{E_m}$	The positional error vector at the actual section tip.	
$\overrightarrow{E_t}$	The positional error vector at the tip-attached marker.	
$\overrightarrow{V_{a/d}}$	The marker offset vector.	
q	The number of subsections within a section of the robot (Navigation).	
f	Feed-in step length (Navigation).	mm
m	Step number of a navigation path.	
w	Weight coefficient of the navigation objective function.	
E_{tip}	The body deviation between the desired and actual configuration of the robot.	mm
E_b	The tip position (absolute) error.	mm
${}^{a}_{b}D$	The minimum Euclidean distance between curve a and curve b.	mm

Chapter 1 Introduction

1.1 Research background

In the 21st century, the fourth industrial revolution (Industry 4.0) brings technological changes in the fields of smart manufacturing, smart sensors, smart automation, artificial intelligence, etc. [1-3]. As an important part of this phase of industrial change, advanced robotics draws more attention from industry to increase productivity, improve product quality, reduce the cost and provide better service to meet customer expectations [4]. In practical applications, several famous industrial robots have been widely used in all walks of life, such as Delta Robots [5], SCARA Robots [6], Cartesian Robots [7], and Articulated Robots [8], etc. These robots are designed with various mechanisms to meet different working requirements.

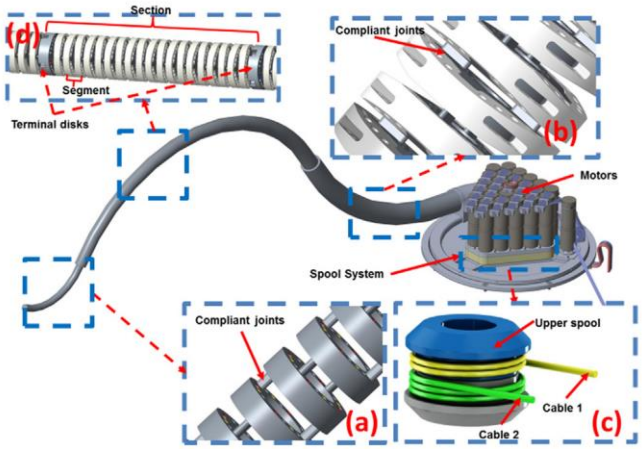


Figure 1-1 Concept drawing of a slender continuum robot, with structure design of robot arm and actuation system [9].

Continuum robots are a kind of robot designed for industrial inspection and maintenance, as defined by Singh and Krishna (2014), a continuum robot is a type of robot that is characterized by infinite degrees of freedom and the number of joints [10]. These robots are generally structured with multiple small diameter sections (Figure 1-1) and benefit from their hyper-redundant and soft structure, enabling them to access and operate in

narrow & dangerous environments, where the working environment is beyond the capability of humans or even traditional rigid-link robots to reach [11].

After decades of development, the technology of continuum robots is gradually maturing, and its application value in in-situ repairing, especially in the high-value-added industry such as nuclear, medical & aerospace, has been verified [12, 13]. However, the hyper redundant manipulator structure and extra-long length of the industrial practical cable-driven continuum robot introduce a new problem in control accuracy. For instance, a short single-section continuum robot could usually have a good performance under kinematics-model-based control while for a slender snake arm (i.e., a robot with a large length/diameter ratio), there exist many unneglected effects such as cable elongation, inhomogeneous friction between cable and disks, self-weight (gravity), etc. under the same condition [14-16], which are always unpredictable and difficult to be compensated without additional sensors. Consequently, more research needs to be conducted regarding the sensing capabilities and controllability of continuum robots, especially in multi-section problems.

1.2 Problem definition

In the current research of continuum robots, a variety of mechanical structure design schemes of the manipulators and the actuation system have been proposed and demonstrated, with corresponding studies regarding the modeling and control method for such hyper-redundant systems (will be presented and discussed in Chapter 2). However, there is still a gap in the research on shape sensing and task-space closed-loop control of a multiple-section continuum robot.

The main goal of this study is to enhance the sensing capability and controllability of a multiple-section continuum robot with spatial motion ability (Chapter 1.3), to enable the robot to work in narrow spaces with barrier-free environments, such as pipework and tunnel, which is often seen in various industry sectors (oil and gas, chemistry, water). The results will be assessed regarding cable displacement control accuracy, operation

repeatability, tip positioning accuracy, path following accuracy, and navigation performance.

1.3 The prototype robot

This research work focuses on cable-driven continuum robots and will be validated on a two-section robot as an example of multiple-section systems. The prototype robot (Figure 1-2) has a twin-pivot arm structure and rotary actuation design (was initially designed by Dr. Xin Dong and Dr. Mark Raffles [15]).

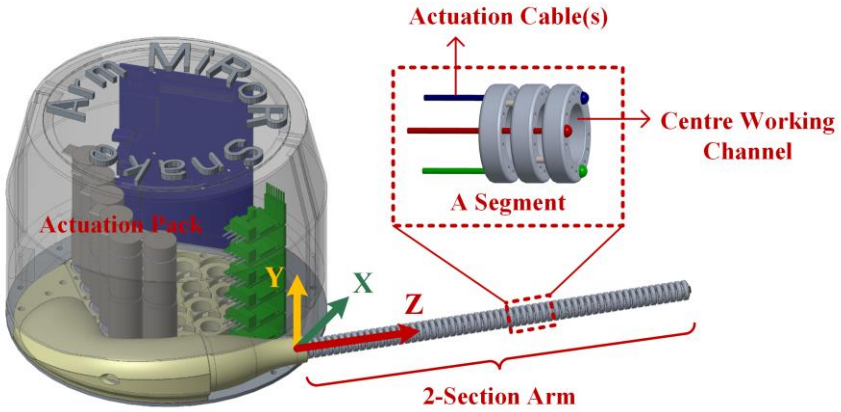


Figure 1-2 The 2-section cable-driven prototype robot.

1.3.1 The twin-pivot robot arm

Different from conventional elastic backbone continuum robots, the twin-pivot continuum robots have a hollow disk design, where the disks are connected by a pair of compliant joints at the edges [15]. **A segment** (Figure 1-3 (b)) consisting of 3 disks and 2 pairs of compliant joints is the primary unit of a continuum robot (Figure 1-2), adjacent pairs of joints are alternating 90 degrees along the longitudinal axis, allowing 3-D motion of the robot. Further, several segments can be assembled to form **a section** (Figure 1-3 (c)), which is the functional unit of a robot with 2 degree-of-freedom, driven by a set of actuation cables to perform functional work (i.e., specifically for a tendon-driven continuum robot). The perpendicularly-arranged-pin segment design (paired joint) increases the torsional stiffness of the robot arm, minimizing the unwanted

deflection as well as improving the load capability of the distal tip. This design also introduces a center working channel in the robot arm (Figure 1-2), thus the tools such as cameras or milling machine spindles can be delivered to narrow space through the channel to work for inspection or repair.

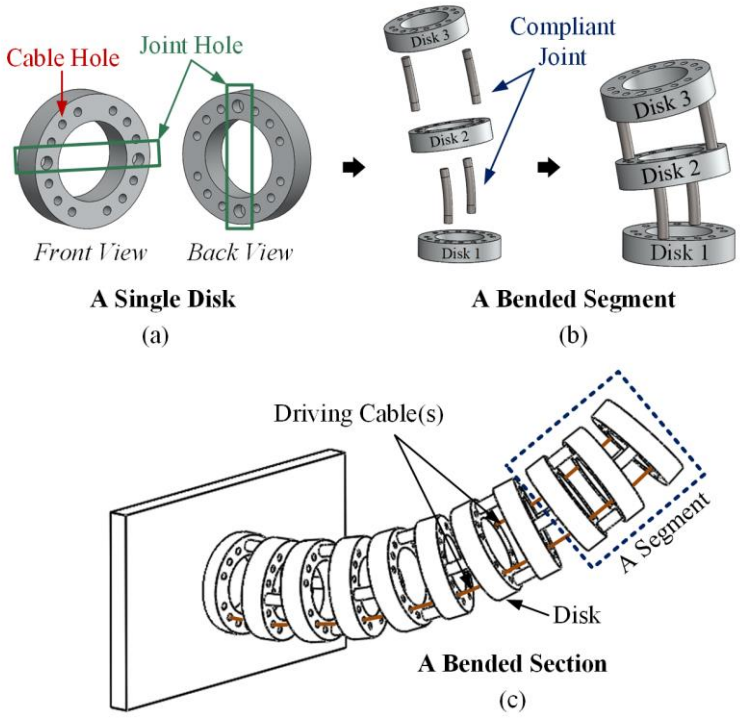


Figure 1-3 The pivot-joint robot design. (a) structure of a disk; (b) structure of a segment; (c) a bended section consists of several segments driven by actuation cables.

For the prototype robot, the disks are made of Aluminium, with 3.5mm thickness, 15mm outer diameter, and 9mm channel diameter. The compliant joints are made of super-elastic NiTi rods, with a 1mm diameter and the single joint buckling load can reach 2207 N. Each section contains 18 disks, with a section length of 120mm and a total weight of 62g (i.e., the net weight without attached sensors or cameras).

Driven by 3 actuation cables, a single section can be bent [0°, 90°] in an arbitrary direction (in a 3-D space). The robot coordinate is defined as (subject to the right-hand rule): the Z-axis is along with the direction of the arm, the Y-axis is pointing upward vertically, and the X-axis is defined horizontally. The robot has a **world coordinate**

defined at the base of the whole robot (Figure 1-2), and each section has a **local coordinate** defined at the base of each section. All the following calculations and modeling are based on this definition of coordinates.

1.3.2 The rotary driving system

In the field of continuum robotics, there are two main approaches for driving actuation cables, which use linear or rotary actuation units. Compared with linear actuators, rotary ones allow compact design and can remarkably reduce the volume of the actuation system [11, 17], which is employed in the prototype continuum robot.

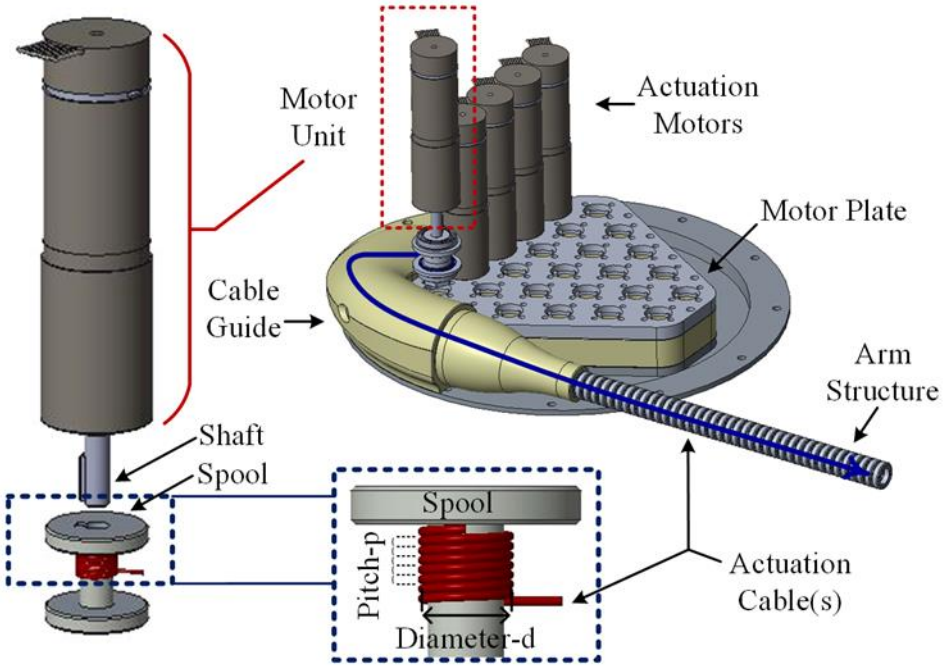


Figure 1-4 The rotary actuation design.

Figure 1-4 demonstrates the inner structure of the actuation pack. Several high-torque motors (4.5 Nm) are arranged on a motor plate, the motor shafts are fixed with some spools where the actuation cables are attached and winded. Each actuation motor can provide 600N actuation force and control the pose of a continuum arm by driving or releasing actuation cables, which serially work through the motor-spool system and the arm structure and are then separately applied to the tip of every individual section. The

spool parameters d (diameter) and p (pitch, also is known as cable diameter) are used to convert actuation motor rotation angle ω_m to actuation cable displacement Δl, and vice versa:

$$\Delta l = \frac{\omega_m}{2\pi} * \sqrt{p^2 + (\pi * d)^2} \quad (1)$$

This calculation can theoretically predict the cable displacement based on the motor rotation measured by the rotary encoder, provided that the cable is properly tensioned, and the pitch is constant between the adjacent revolutions, which is commonly used in actuation-space and joint-space control of continuum robots.

1.4 Objectives of this study

A comprehensive robot sensing and control scheme needs to be proposed and implemented, to improve the controllability of a multiple-section cable-driven continuum robot with a rotary actuation system design, exploring the possibility of the robot for applications in extreme environments.

The objectives of this project can be classified into 4 subtasks:

- Conduct an in-depth control error analysis of the current continuum robot system for proposing potential improvements, which can be assessed from these aspects: 1) cable displacement control error within joint-space; 2) positioning error within task-space.
- Develop an integrated vision & sensing system for continuum robots to feedback on real-time control error both in joint-space and task-space, which demonstrates a feasible and reliable sensing capability to enable closed-loop control. An outward-looking sensing approach is also needful to observe the working environment and execute environment interactive tasks.
- Develop a closed-loop controller based on the above sensing method, working to converge the errors both in low-level and high-level control, which shows a demonstratable improvement in robot tip-position control regarding accuracy

and repeatability. Specifically, this controller is supposed to **keep the tip-positioning error within millimeters**, thus precisely operating the robot in **target working environments** presented in Chapter 1.2

- Propose an appropriate adaptive navigation strategy for a continuum robot and equip the robot with abilities of obstacle detection and avoidance. Thus, the robot can be navigated into a confined space and take on more complex tasks.

1.5 Structure of the thesis

Chapter 2 reviews the state-of-the-art development in the field of continuum robots, a wide range of literature and academic publications are cited to discuss the mechanical design, modeling, control strategies, and sensing solutions for this kind of robot with snake-liked shape and hyper-redundant structure. Finally, the potential opportunity for further research is presented concerning sensing capability and integrated control schemes for continuum robots.

Chapter 3 detailly introduces the structure design and basic modeling & control methods for the prototype robot system utilized in this research work. The 2-section tendon-driven continuum robot employs the twin-pivot arm design, rotary (motor-spool) actuation system design, and PCC-based kinematics, introducing several challenges in precision control, which will be discussed at the end of this chapter.

Chapter 4 presents a hybrid sensing system designed for continuum robot systems as well as a two-level closed-loop controller developed based on the new sensing solution. Regarding the sensing solution, the design and integration of the linear encoders and camera & marker arrays are explained minutely, with the working principle in joint-space and workspace error monitoring. To implement these sensing devices in robot control, a two-level closed-loop controller is also proposed, where the low-level controller employs the readings from the rotary and linear encoder to eliminate the control error in the joint-space, and the high-level controller based on a model-less control method aims to converge the tip-position error in the work-space. Under the

work frame of the two-level closed-loop controller, this system aims to improve the control accuracy both in joint space (for actuation cable displacements) and workspace (for tip positioning). Finally, a series of experiments were conducted to validate the sensing system and the robot controller in terms of the feasibility of visual tracking, error compensation rate for joint space control, and tip-positioning accuracy in both reference and world coordinates.

Chapter 6 discusses a vision-aided environment-interactable navigation strategy proposed for the continuum robot. The navigation is divided into 3 stages, the work frames of path pre-planning, navigation process, and target detection are introduced in turn. Finally, the path-following performance of the distal tip and the trajectory-following performance of the snake body is examined by an open-space navigation test, followed by a working demo to demonstrate the compatibility of the robot to avoid obstacles and go through obstructed channels. Besides, a remote controller (App on Android devices) is developed for the robot to make the robot can be operated safely in extreme environments.

Chapter 7 concludes the research work presented in this thesis, including academic and engineering contributions, as well as making some suggestions for further study in the field of continuum robots.

1.6 Highlights and contributions

In this part, the highlight and contributions of the thesis are presented, including a comprehensive sensing solution for the multiple-section continuum robot system, a novel two-layer control strategy based on the new sensing solution, and a novel vision-aided adaptive navigation scheme with remote controllability.

1.6.1 Comprehensives sensing solution for a continuum robot system

A hybrid sensing system is designed and applied on a two-section cable-driven continuum robot prototype, which includes a dual displacement sensor system to work

in joint space error compensation and a self-carried vision system to work in work-space for tip tracking & environment observing.

- The dual displacement sensor system is formed by a group of rotary encoders built into the actuation motors and a set of linear encoder based measurement units equipped at the proximal end of the robot arm (i.e., between the actuation pack and the robot arm, will be demonstrated in Chapter 4). These two kinds of sensors will work cooperatively to indicate the displacement control error in the actuation pack and predict the cable elongations along the arm, providing feasible feedback for joint space control (low-level control).
- The self-carried vision system consists of several paired observe-reference camera & marker arrays and a head camera. The camera & marker array are formed by miniatured probe cameras and specially patterned visual markers, where the cameras provide section-to-section tip tracking for the continuum robot by detecting the assigned markers. According to this information, the controller can be developed to converge the control error of robot pose and improve the accuracy of tip positioning. Besides, the head camera serves to observe the surroundings and locate potential obstacles against the robot (the outward-looking).

1.6.2 A novel two-layer control strategy based on the new sensing solution

This closed-loop controller is developed based on the sensing system introduced above and is divided into a joint-space cable displacement controller and a work-space tip position controller (which will be introduced in Chapter 4).

- The joint-space controller relies on the cable displacement feedback from the dual displacement sensor system and a novel two-stage cable displacement control approach. This new method split the cable displacement control into 2 stages, where the first stage evaluates the errors and the second compensates

for the errors, aiming to minimize the error caused by unideal winding (on the motor-spool actuation mechanism) and cable elongation.

- The work-space controller employs the tip-tracking from the camera & marker arrays, implemented with a novel indirect measurement based tip-adjustment strategy, which is a model-less method to converge the positioning error of robot tip(s) and drive the robot tip to a target region. This controller can correct the robot pose section-by-section iteratively and achieve high-accuracy tip positioning and path following.

1.6.3 Novel vision-aided navigation approach for continuum robot

When the precision tip position control is achieved, a navigation strategy is proposed to navigate the robot in some real working scenario. Within this, a led-by-the-tip method is introduced to plan the motion of a continuum robot, under this work frame, the distal tip will go forward according to a pre-planned path and the robot body will snake its way along the trajectory of the distal tip. Besides, the function of the head camera is also developed to monitor the working environment and detect environment features, which can help with obstacle avoidance and object measurement.

For the purpose of safe operation in hazard environment, an application based on an Android tablet PC is also developed to mimic a joystick and control the robot remotely. This App is programmed in Android Studio with JAVA and communicates with the main control robot by TCP-IP communication protocol. The functions include controlling the pose of an individual section via bending angle θ and direction angle ω, navigating the robot by a pre-planned path forward or backword, etc.

Chapter 2 Literature Review

As introduced in Chapter 1 , the technology of continuum robots is gradually maturing whose application value has been proved in accessing confined space & hazardous environments to execute complex operations such as in situ repairing of aero engine, nuclear station prospection, medical surgery, etc. In this chapter, a wide range of literature and publications are reviewed, and state-of-the-art development in the field of continuum robots will be studied and summarized regarding mechanical design, modeling approach, sensing capability, control strategy, application scenario, etc. Followed by a brief discussion of the research gap and potential research opportunities in this field.

2.1 Continuum robot design

The most prominent feature of continuum robots is the hyper-redundant and compliant structure [18], which not only distinguishes them from conventional rigid-link robots or machine tools but also allows them to access narrow or confined spaces and extreme working environments. Generally, continuum robots can be classified according to the actuation mechanism and mechanical features of the manipulator. The actuation system of a robot can be pneumatic, tendon-driven (cables or rods), fluidic, ferromagnetic, EAP-based, concentric tube-based, and hybrid actuation design; and the manipulators can be sorted by functional space (planner or 3-D), number of sections (single or multiple), shape design (with/ without a backbone), and materials [19]. In this section, the design of continuum robots will be discussed regarding the actuation mechanism, and all the design schemes will be summarized and listed.

2.1.1 Tendon-driven continuum robots

Tendon-driven actuation is one of the main categories of driving system designs of continuum robots. This kind of robot always employs cables or rods (Figure 2-1) [20-

22], driven by remote actuation such as a servo motor singly or in pairs, to adjust the shape or curvature of a continuum manipulator. This type of remote actuator can be installed at the far end and drive the tendons by a transmission system, which allows the compact design of the manipulators [23].

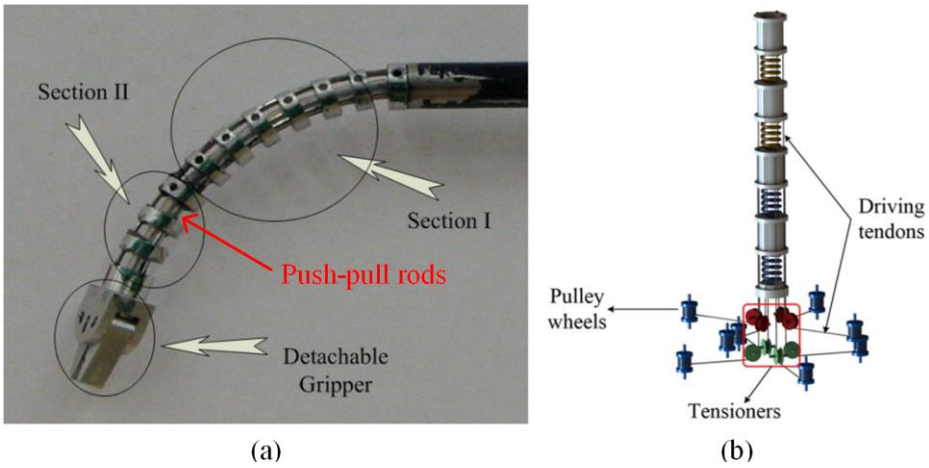


Figure 2-1 Examples of tendon-driven continuum robots. (a) A snake-like robot actuated by push-pull rods [22]. (b) A continuum robot driven by actuation cables [24, 25].

The manipulator could be classified according to backbone design. The center backbone consists of universal joints, springs, flexible rods, etc., and the manipulators are formed by compliant center backbones and spacer elements such as disks and plates working to guide and terminate the driving tendons, the tension force can be applied to the robot structure to affect the bending of the backbone. Some other continuum robots deprecate the center backbone design but use other solutions such as employing side-located joints, one-piece (monolithic) arm structure, filling materials, etc. To enhance the performance of such robots, some designers also employ hybrid manipulator design, which promotes the strengths and offsets the weaknesses of each mechanism.

Regarding remote actuation, there are two main approaches for driving the actuation tendons, which use linear or rotary actuation units (Figure 2-2 (a) and (b)). By comparison, rotary actuators can remarkably reduce the volume of the actuation system while linear actuators can provide better feasibility regarding tendon displacement control accuracy.

The control inaccuracy within the tendon-driven robot system is mainly introduced by friction, tendon backlash[26], slack of tendon, tendon coupling (i.e., for multiple-section robots) [27], etc. Further studies need to find methods to avoid these problems, for instance, the design of a pully-guided wire-driven system can significantly decrease the friction between the actuation tendons and the rigid parts of a robot [28].

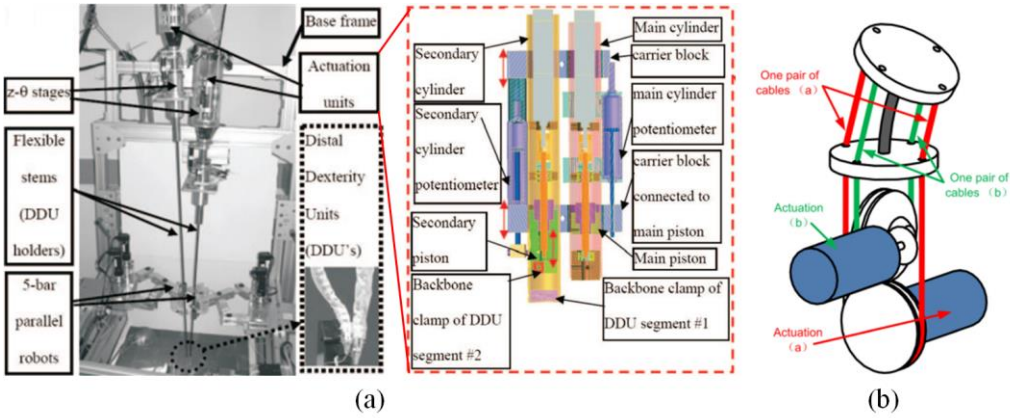


Figure 2-2 Examples of the actuation systems of tendon-driven continuum robots. (a) The linear actuation system employs linear motors for driving the tendons [17]. (b) The rotary actuation unit utilizes rotary motors and spool systems to drive the tendons [15].

2.1.2 Pneumatic continuum robots

Pneumatic actuation is another mainstream in continuum robot design. This category of robots always adopts the concept of 'air muscle'- a pressurized-air-filled elastic body whose deformation is pre-defined by external constrains (Figure 2-3).

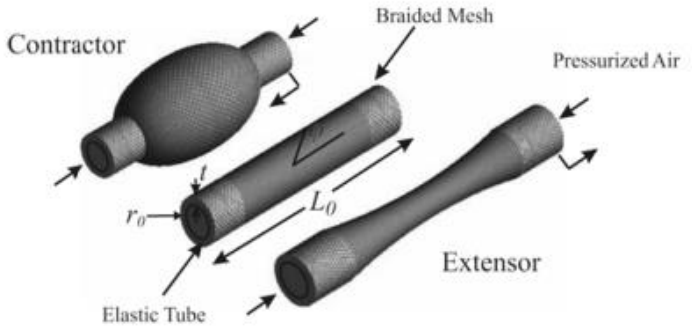


Figure 2-3 Examples of air muscles composed of elastic tubes and constraints [29].

The length, configuration, effort force, etc. of a manipulator are generally controlled by the pneumatic/ hydraulic pressure via pumps or pressurizers. Referring to the arrangement of pneumatic elements, the pneumatically actuated continuum robot can be classified into parallel-arranged actuators (parallel tubes or bellows), helically wrapped actuators (the wound tube actuator, WTA), and everting actuators.

2.1.3 Other types of continuum robots

Exception for the tendon-driven and pneumatically actuated continuum robots, there are still some other actuation approaches that may not be widely used, such as concentric-tube robots, fluidic robots, ferromagnetic continuum robots, electro-active polymers (EAP) based robots etc. Additionally, in some cases, these actuation methods can be combined to improve the performance and enhance the capability of a continuum robot.

Finally, the common designs for continuum robot systems are summarized below. **In the following sections, this thesis will focus on the cable-driven continuum robot and relative modeling and control approaches.**

Table 1 General types of continuum robots

Tendon-driven continuum robot		features
With centre backbone	Universal joint backbone [30-32]	- High flexibility - Low stiffness - Inevitable effect of twisting (self-weight & end payload)
	Spring backbone [21, 24, 25, 33-36]	- Extensible and compressible - High flexibility - Low stiffness - Hard to control (low control accuracy) - Inevitable effect of twisting
	Flexible rod backbone [15, 22, 37-39]	- Allow slender design (minimize the diameter/ length ratio) - Can achieve various stiffness by the material selection - Inevitable effect of twisting
	Hybrid backbone [21] [31, 40] [32]	- Combining the advantages of various types of backbone design

Without centre backbone	Side located joints [15] [41]	- Minimize twisting effect (around the arm axis) - Preserving a center channel in the manipulator (for delivering tools in confined space)
	One-piece (monolithic) arm structure [42, 43] [44]	- Always made of soft materials (rubber, silicon, latex, etc.) - Easy to be prototyped - High flexibility
	Particle-filled structure [45, 46]	- Jamming mechanism, with variable stiffness (adjusted by air pressure) - Self-lockable
Hybrid manipulator design	Driving cable & push rod [20]	- Higher rigidity and stability - Minimizing the twisting effect - Increasing the load capability
	2-D and 3-D sections [13]	- Composed of self-lockable planner sections and 3-D tip sections. - Increasing the payload - Limited applications (with typical lock position).
	Rigid link & flexible joint [47-49]	- Increasing the stiffness over the length of a manipulator - Losing the flexibility to some extent.
	Telescoping arm structure [50]	- Enlarging the workspace - Reducing the path deviation errors - The inner robot can be a rigidizing device
Pneumatically actuated continuum robot (Air muscle)		features
Parallel-arranged actuators	Parallel tubes: FMA [51], OctArm[29, 52-54], STIFF-FLOP Surgical Manipulator [55]	- Common structure: 3 parallel-arranged reinforced tubes - Cover with meshes to precent weak area while bending
	Parallel bellows: BHA[56, 57], CBHA[58-60] [61]	- Improved version of the parallel-tube robot - Higher stiffness and extensibility - High-complexity structures, usually made by additive manufacturing
Helically wrapped actuators	The wound tube actuator (WTA) [62, 63]	- Consists of restricted parts and unrestricted parts - Only the unrestricted parts response to air pressure change

Other structure design concept	Octopus anatomy-inspired robotic arm [64], Honeycomb-like pneumatic robot [65, 66], Vine-like robot [67, 68]	- For octopus anatomy-inspired robot: enabling radial and longitudinal expansion simultaneously - For vine-like robot: self-growing (adjustable length)
Other actuation method		features
Concentric-tube robots	[69, 70] [71]	- Assembled by pre-curved super-elastic tubes made from shape memory alloy (SMA) - Small size - High dexterity - Having good potential in medical-related small-scale applications with low force.
Fluidic robots	[72] [73] [74]	- Similar to the pneumatical continuum robots - Driven by fluidic drive cylinders
Ferromagnetic robots	[75-78]	- Small-scale (in mm/ sub-mm) - Composed of soft polymer matrix and embedded hard-magnetic particles - Steered by a magnetic field - High steerability - Remote controllability - Having potential for minimally invasive treatments
EAP	[79, 80]	- Similar to ferromagnetic continuum robots - Structured by electroactive polymer and electrode - Driven by an electric field - Small scale (in mm)
Hybrid-actuated continuum robot		features
pneumatic-cable driven continuum robots	the AirOct [81]	- Combining features of multiple types of robots - Improving the performance of a robot system
	the KSI [82, 83] tentacle manipulator	
Tendon-driven magnetic robot	[84, 85] [86]	
Tendon-driven concentric tube robot	[87]	

2.2 Continuum robot modeling

Modeling is the essential work for continuum robots' control and application, which maps the relationship between each operation space of a robot. Generally, there are four operation spaces in continuum robot modeling and controlling, which refer to actuation mechanism (actuation & joint space), arm structure design (configuration space) and operating target (task space) as listed in Figure 2-4. The modeling approaches are basically divided into two categories, kinematic modeling, and dynamic modeling, the former describes the robot behavior under steady-state conditions (force equilibrium), while the latter is the study of motion.

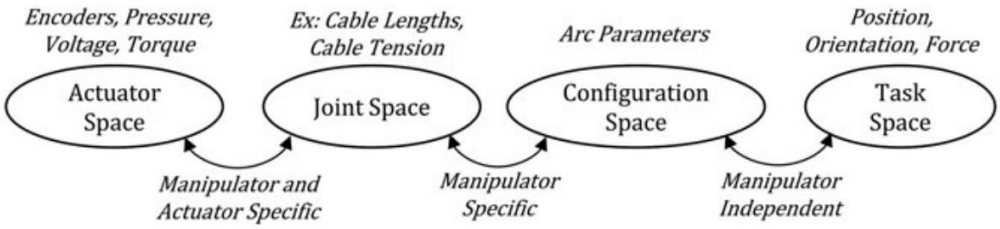


Figure 2-4 Operating spaces of a continuum manipulator and their definitions [19].

2.2.1 Kinematics model

Differing from conventional rigid-link robots, continuum robots have significantly more flexibility since the hyper-redundant design, which increases the difficulty of establishing an accurate model to describe its behavior. In the previous research, there are mainly two basic models applied, the PCC and D-H models.

- Piecewise constant-curvature (PCC) model

The constant-curvature assumption may be the most widely used kinematics model for various types of continuum robots, which describes each section of a robot manipulator as a spatial curve with key parameters of arc length, bending curvatures, and bending direction [18], thus the tip position & orientation can be obtained by geometry calculation (i.e. spatial transformation).

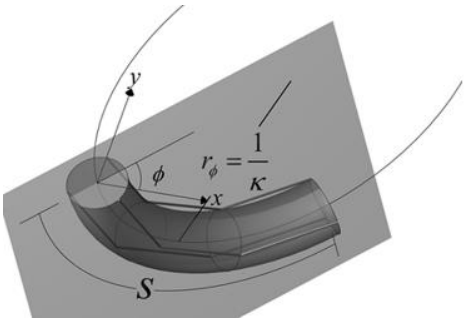


Figure 2-5 A constant curvature geometry [88], where k is the curvature of the curve, ∅ is the rotation if the curve and s is the arc length.

- D-H method (Denavit-Hartenberg)

The core idea of the D-H method is to create a group of virtual links on a continuum robot, to mimic the conventional rotation and translation links robots. In the 2-D problem, the robot arm curvature could be modeled as a three-joint robot with a pair of coupled rotation joints at the beginning and the end of the curve and a middle prismatic joint along the curve direction (Figure 2-6).

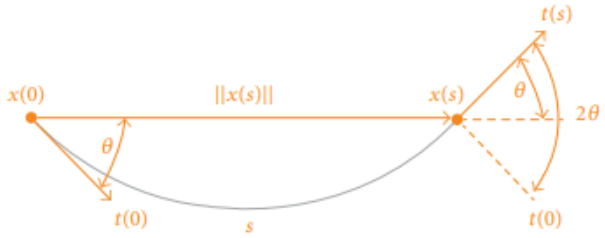


Figure 2-6 2-D H-D geometry a constant curvature continuum robot (single section) [18].

A 3D spatial curve could be simulated by adding an extra pair of rotational joints (representing the spatial rotation) to the planner model, thus, forming a 5-joint virtual robot (Figure 2-7). Since the 2 groups of paired rotational joints are coupled (identical), this robot totally has 3 degree-of-freedom during calculation (i.e., two rotations and one translation).

Eventually, for a single section (the ith section), the results of PCC and D-H methods shear the same transformation matrix in the section plane:

$$T_i = \begin{bmatrix} \cos(s_i k_i) & -\sin(s_i k_i) & 0 & (\frac{1}{k_i})(\cos(s_i k_i) - 1) \\ \sin(s_i k_i) & \cos(s_i k_i) & 0 & (\frac{1}{k_i})\sin(s_i k_i) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

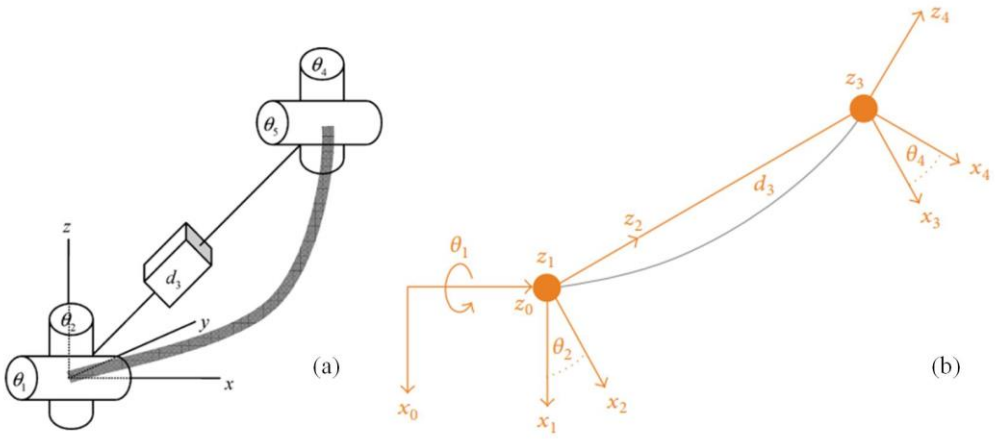


Figure 2-7 3D D-H geometry of a constant-curvature continuum robot (single section). (a) The virtual 5-joint robot [88]. (b) Geometry representation of the robot [18].

Further, the 3-D multiple-section problem can be solved by chain calculation of each individual section.

- Learning-based method

With the development of the sensing ability of continuum robot systems, learning-based control, and modeling methods are increasingly appearing in relative research. This method formula the non-linear relationship between input and output quantities through neural network training as shown in Figure 2-8. The neural network is composed of an input layer, a hidden layer, and an output layer. The layers contain several neurons storing data, and each neuron is propagated by the previous layer via a function with parameters of weighs w, bias b and activations a.The thought of NN training is similar to data fitting, the relation between input and output data is famulated by finding optimal parameters with the least error.

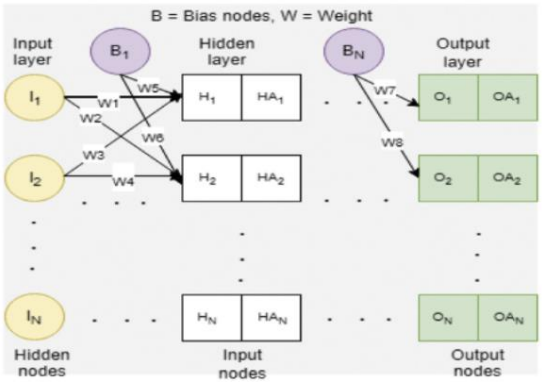


Figure 2-8 An example of typical neural network architecture [89].

For kinematic modeling, the general working process includes problem formulation, data collection, and parameter fitting, and the method has been implemented in both forward and inverse kinematics [90-92].

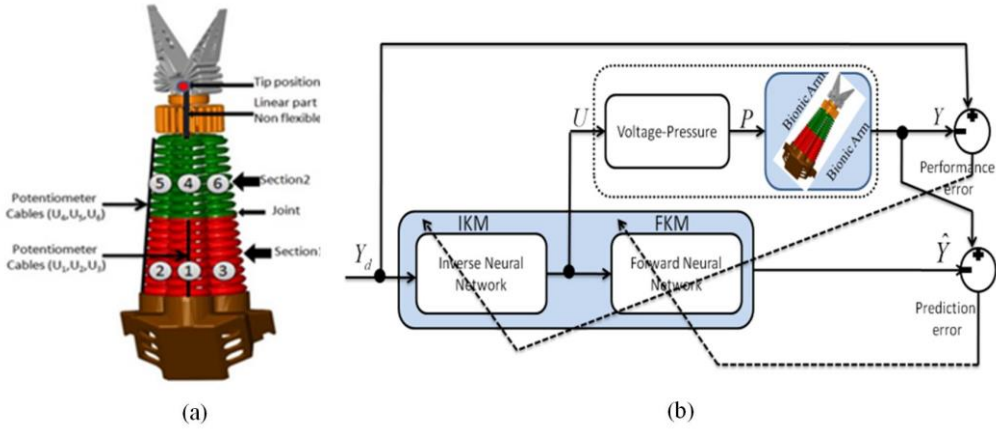


Figure 2-9 Neural network-based model learning of CBHA [90, 91]. (a) The robot configuration. (b) Inverse Neural Network learning.

The prototype robot in [90, 91] is a pneumatic continuum robot CBHA introduced previously, this robot has 2 sections, and each section is composed of 3 pneumatic tubes, some potentiometer cables are attached to the robot to measure tube length (Figure 2-9 (a)). In the case of computing forward kinematics, the object is to map from tube length to the tip position. Herein, the input layer consists of 6 measurements from wire-potentiometers (U_1 to U_6) and the output layer is the 3 Cartesian coordinates of the tip. Two types of neural networks MLP (Multilayer Perceptron) and RBF (Radial Basis Function) are used separately while no prior information about the robot's kinematics is

provided. The forward kinematic models show good performance when applied to model-based control tasks, both the error is smaller than the geometric model. Based on the forward training, inverse training is also conducted (Figure 2-9 (b)) under the supervised learning method. Since the redundancy of the system (i.e., 3 inputs and 6 outputs), the training procedure includes the previously obtained forward kinematics as prior information which works in serial with the inverse training set for mutual validation.

To sum up, the traditional kinematics model such as D-H method and PCC method can theoretically predict the robots' behavior while the accuracy is not guaranteed under external conditions. The advanced non-linear method such as the learning-based method is much more accurate, however, it is a time-consuming process to train the robot. For further research, model-less or model-free methods show their potential.

2.2.2 Dynamics model of continuum robots

Dynamic modeling of a continuum robot system is much more complex than the kinematics one since it considers the interactive force and describes the motion of a system in the time domain. To build a dynamic model, the corresponding kinematic model and dynamic formulation are required. For example, a method based on the constant-curvature kinematic model and Euler–Lagrangian equation is raised by Tatlicioglu and Walker (2007), to describe the dynamic behavior of a planner extensible pneumatic robot [93, 94]. Similarly, the principle of virtual power [95] and Cosserat rod theory [96, 97] could also be applied in continuum robot systems to deduct dynamic formulas. FEA (Finite Element Analysis) is another available option for elastic body continuum robots (such as the pneumatic or hydraulic continuum robots) to establish a kinematic or dynamic model since the uniform or continuously varying structure and material properties along the body [98]. In the referred article, the data collected from FEA simulation is used to pre-train a learning-based model and got a satisfactory results in the experiments.

In conclusion, there are different forms of dynamic models to formulate various types of continuum robots, which is always a complex time-domain equation. In this research work, dynamic behavior is not considered, attention will be paid to the kinematics of continuum robots.

2.3 Sensing solutions for continuum robot (shape sensing & tip tracking)

The hyper-redundancy of a continuum robot leads to several condition-dependent uncertainties and errors, differing it from theoretical modeling or prediction in the practical system to some extent. Consequently, the conventional theoretical kinematic or dynamic model-based controller introduces non-negligible errors in motion control. To account for this, employing sensors and developing new controllers could be the solutions to minimize uncertainties and errors and improve the controllability of continuum robots. The control of the continuum robot can be classified into four operating spaces: actuation space, joint-space, configuration space, and task space [19] (i.e., For a cable-driven continuum robot, the actuation space and joint space control refer to the actuation motor and actuation cables respectively. The configuration space indicates the robot arm arc parameters such as the bending angle (θ) and direction angle (φ) of each section. The task space refers to the position, orientation, and force control of the robot tip in the global coordinate, these terms will also be introduced in detail in Chapter 3). Various approaches have been developed to measure these robot parameters and provide feasible feedback to achieve closed-loop control of the arm.

Generally, sensing methods can be divided into two categories: contact and contactless. The contact sensing methods always employ embedded sensing devices such as resistive sensors, piezoresistive sensors, etc. while contactless sensing relies on external tracking devices, for example, the infrared or visual tracking system.

2.3.1 Contact sensing solutions for shape sensing

Contact sensing methods usually pertain to built-in (embedded) sensors, by employing self-sensing materials to build a robot or implementing commercial transducers.

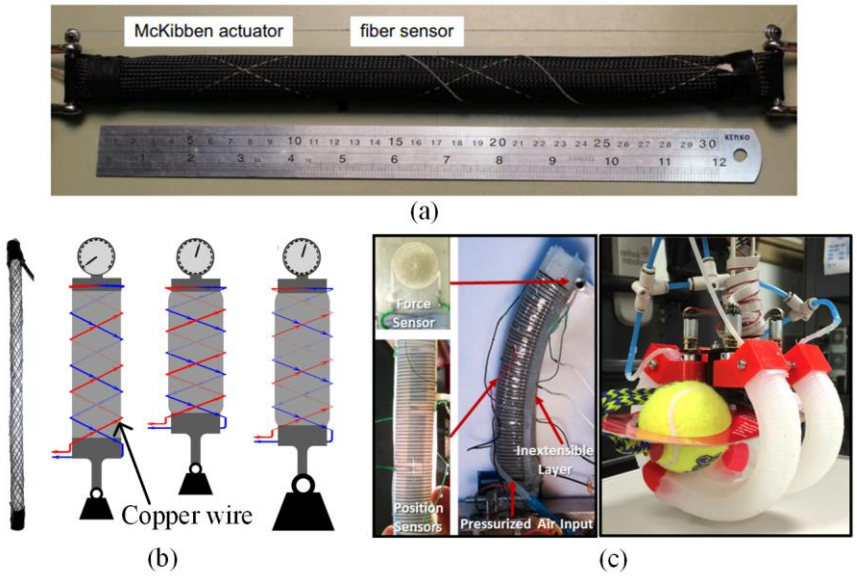


Figure 2-10 Examples of stretchable conductive material applications for building continuum robots. (a)A McKibben actuator, an example of electro-conductive yarn-based shape sensing [99]. (b) The Smart Braid, an example of inductance-based shape sensing [100]. (c) A soft pneumatic actuator (SPA), an example of liquid metal-based shape sensing [101].

- Self-sensing material (stretchable conductive materials)

Some sensors are developed according to variable material properties (e.g., the resistance change with the shape), which mainly refers to stretchable conductive materials [102]. These materials are usually integrated with the robot structure or used to build the main body of a robotic arm, equipping the system with proprioception sensing ability.

Figure 2-10 gives some examples of applications of stretchable conductive materials. The first prototype (Figure 2-10 (a)) is a McKibben pneumatic actuator. Here, a fiber sensor (i.e., conductive yarn, the solid line in the figure) is attached to the membrane of the actuator. The two ends of the extensible conductive yarn are fixed with the actuator; thus, the circumference of the cross-section can be predicted by resistance change. The concept of the Smart Braid (Figure 2-10 (b)) is to replace the normally used nylon fiber

with conductive copper wire as reinforcing fibers on a pneumatic actuator (i.e., the red and blue helices in the figure). The two sets of helices (right-handed & left-handed) are embedded in silicone and powered, when the shape of the pneumatic actuator changes, the inductance of the circuit as well as the resistance of the conductor also change. Figure 2-10 (c) introduces a position sensor based on liquid metal (i.e., the eGaIn strain sensor). This sensor is directly embedded into the surface of a soft robot by casting micro channels into the silicon layer and injecting the liquid metal into the channels. Similar to the first example, the curvature of the robot is predicted by measuring the resistance change of the sensing material.

Apart from the materials listed above, stretchable conductive materials also include ionic liquid (NaCl solution) [103, 104], ionic hydrogel [105], CBF sensors [106], or even commercial Flex bend sensors [107]. This type of sensor is mostly applied on elastic (soft) body robots made of rubber, silicon, or polymer, (i.e., as the pneumatic or fluidic continuum robot), with a single-section structure. They are not suitable for shape sensing in slender arms as errors accumulate along the length of the measurement and the results are always with considerable noise.

- Displacement sensors for shape sensing

Displacement sensors such as string potentiometers (metallic cables) and linear or rotary encoders are always employed in a continuum robot system. These devices usually work in two ways: 1) provide feedback displacement of actuating cables or pneumatic tubes to achieve closed-loop control and improve control accuracy in joint-space; 2) measure the tendon displacement or elongation of the arm to predict robot configuration through geometric calculation (i.e., in this case, additional sensing cables are always attached to a robot, excepting the actuating cables). These sensors are commonly mounted in the actuation pack, and the arm shape or tip position is estimated by the kinematics models. However, this indirect shape estimation has inherent errors introduced by mismatches between the theoretical model and the actual arm shape [16].

Figure 2-11 (a) shows a typical example, a group of potentiometers are embedded on a cable-driven continuum robot to feedback arc lengths of all the sections. Similarly, prototype (b) is equipped with some additional passive cables and estimates the shape of a two-section robot by cable displacement via a virtual powder model. Alternatively, Figure 2-11 (c) provides another option for linear encoders. In this case, the linear encoder chips are mounted on the motor shaft to control the linear displacement of actuation cables. The last example (Figure 2-11 (d)) is a continuum robot structured by pneumatic bellows, similar to prototype (a), some potentiometer cables are attached to measure tube elongations and estimate the position of the end effector.

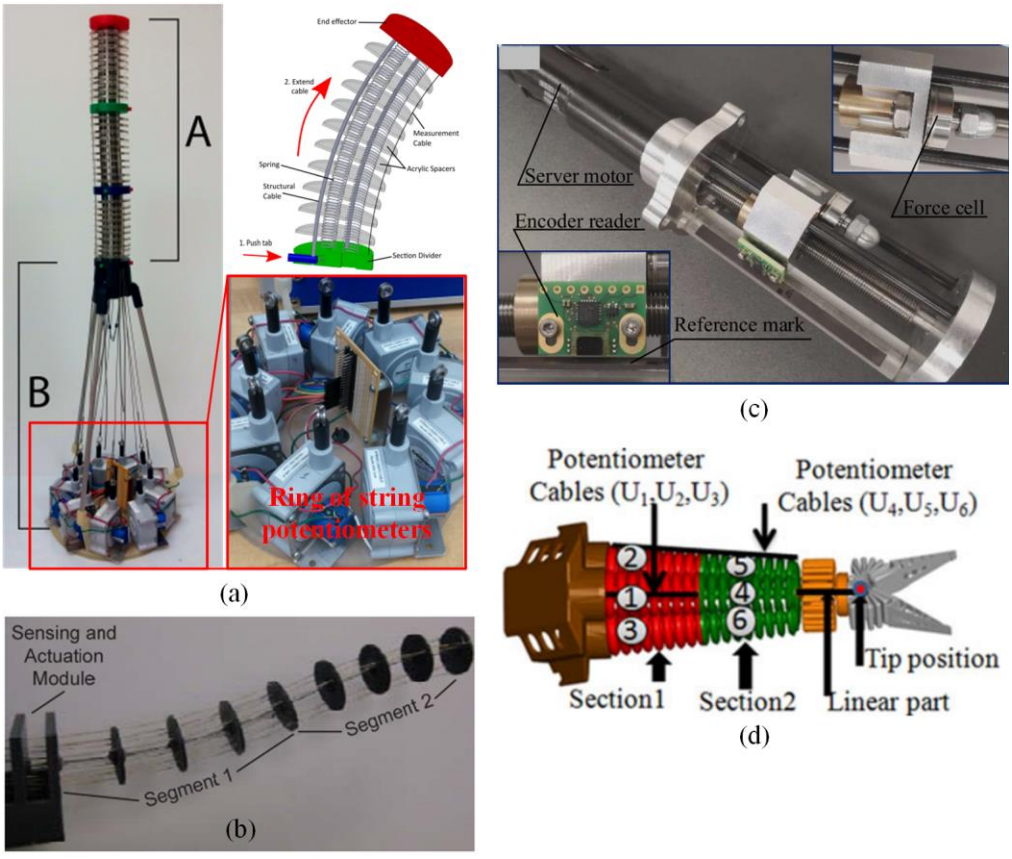


Figure 2-11 Examples of displacement sensors for measuring tendon displacement. (a) An example of applying potentiometers on a cable-driven continuum robot to control tendon displacement [36]. (b) An example of using passive cable displacements to estimate robot shape [108]. (c) An example of using a linear encoder to control cable displacement [13, 16]. (d) An example of a pneumatic robot equipped with potentiometer cables to estimate the position of the end-effector [58].

- Optical sensors for shape sensing

Optical sensors usually take advantage of the photoconductivity of materials, to detect the variations of emitted light (e.g., phase, frequency, or intensity) after passing through the deformed light transmission medium. Within which, optical fibers are a kind of widely used strain (elongation) sensor as shown in Figure 2-12 (a) & (b). In both examples, the fibers are integrated into the surface of a continuum robot, the elongation is evaluated by the light intensity measured at the proximal end. This sensor is highly deformable, however, similar to the displacement sensor, it is also an in-direct shape estimation method that has inherent errors.

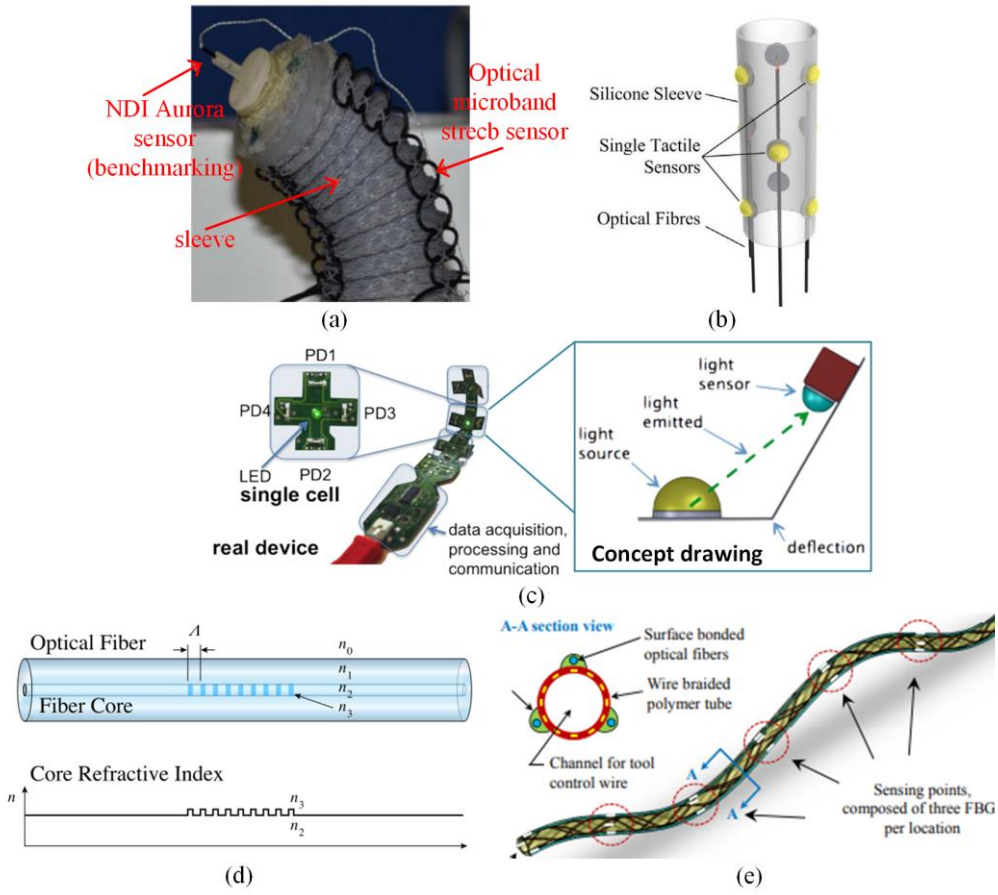


Figure 2-12 Examples of optical sensors. (a) A macro bend sensing system made of optical fiber [109]. (b) A sensor sleeve design containing tactile sensors and optical fiber [110]. (c) A deflection sensor composed of LED and light sensor [111]. (d)The inner structure of FBG sensors. (e) An example of FBG shape sensor application [112].

FBG (fiber Bragg grating) sensors (Figure 2-12 (d)) are advanced versions of optical fiber, since multiple FBG units can be fabricated on the same fiber to detect the local

strain or pressure of each sensing point [102]. Thus, the robot shape can be directly rebuilt by integrating detected local elongation (strain) along with the arm structure, which shows great potential in developing a compact and high-performance 3-D shape detection system as demonstrated in Figure 2-12 (e) and [113-115]. This type of sensing system is more expensive than the other commercial ones (i.e., since the fabrication process and the signal processer).

Figure 2-12 (c) provides another concept to employ optical signals. Herein, an LED light and a photodiode are fixed on the bending parts separately, the physical or virtual deflection angle can be determined by the light intensity received by the photodiode since the typical bell-shaped angular intensity profile of the emitted light.

2.3.2 Non-contact sensing solutions

On the other hand, contactless shape sensing and tip tracking always relies on external measurement devices, these devices are commonly fixed in the working environment of a continuum robot with a known position, observing the robot from outside.

- EM (electromagnetic) tracker

An Electromagnetic (EM) tacker is a commonly used equipment in small-scaled operation (Figure 2-13 (a)) [39]. This system consists of several magnetic sensors attached to the tips of the sections of the arm and a processer tracking the sensors in an artificial magnetic field generated by a transmitter. Both position and orientation measurements are provided at high accuracy, based on the magnetic field variation, with a position error of 1.4 mm RMS and an orientation error of 0.5°RMS [116]. However, limited by its working principle, any ferromagnetic material in the robot or near the field can significantly influence the precision of the measurement.

- Vision & Imaging system

By contrast, external vision systems are more feasible for large-scale applications. This type of measurement system can be configured by X-ray (Monoplane or biplane

fluoroscopy) [117, 118] infrared cameras [13, 57], depth cameras [119], industrial cameras [120, 121]or even general webcams (with special visual markers) [37].

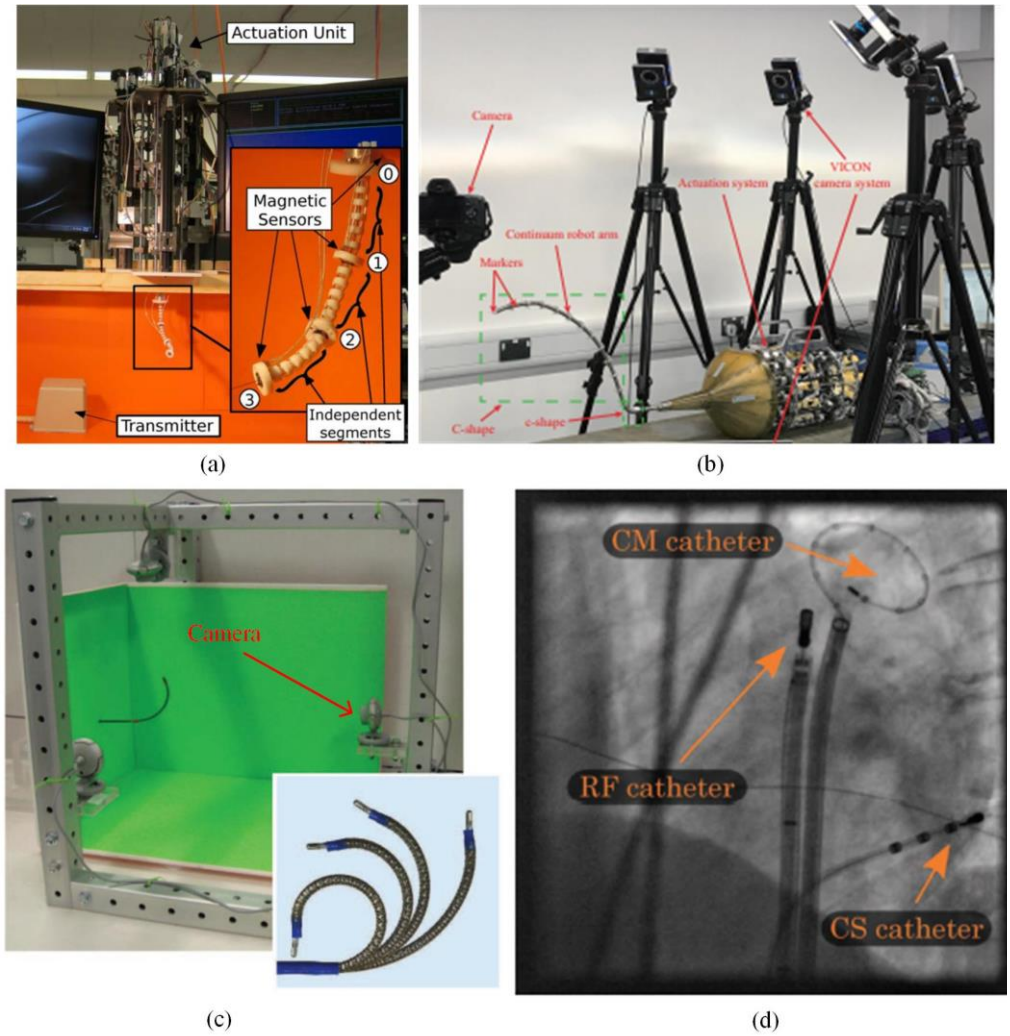


Figure 2-13 Examples of EM tracker and vision & imaging systems for measuring the configurations of continuum robots. (a) Application of EM tracker on a multiple-section continuum robot [39]. (b) Application of VICON system (infrared cameras) in tip-tracking [13]. (c) Shape sensing by stereo camera system [122]. (d) An example of using the x-ray to reconstruct the configuration of a medical robot [117].

Figure 2-13 (b) shows the VICON system (grouped by several infrared cameras) used in a tip-tracking task. Herein, some reflective markers are attached to the sensing points (always the tip(s)) of a robot, to reflect the infrared light emitted from a group of calibrated cameras. Then, the stereo camera system can recognize the markers and provide position feedback relative to a pre-defined world coordinate. This kind of vison-

base system can achieve high accuracy (usually in submillimetre) and is an ideal shape-sensing method for open-space tasks. The general cameras work similarly (Figure 2-13 (c)). After calibrating the camera array, the position and orientation of some feature points (i.e., recognized by some feature detection algorithms) can be obtained by extracting information from a group of camera views. Overall, the external measurement methods can capture robot shapes precisely, eliminating the error in calculation compared to contact sensing. But it presents difficulties when applied in confined spaces since the view of the cameras would be blocked by obstacles.

X-ray is another type of vision-based method that allows intraoperative shape sensing (Figure 2-13 (d)). This type of method usually fused image data with a kinematic model to detect the 2-D or 3-D shape of a small-scale continuum robot tool [12]. However, X-ray fluoroscopy may be only suitable for medical applications and is hard to apply for engineering projects.

2.3.3 Other sensing methods

Except for the general sensing methods listed above, there are other novel sensing methods that are not commonly used but published in previous research works, Figure 2-14 shows two examples. Figure 2-14 (a) demonstrates a solution based on magnetic sensors and permanent magnets fixed on a continuum robot as shown in the picture. Herein, the magnetic flux density of the magnets is detected by the paired magnetic sensor to evaluate the local bending angle of a robot arm, then, based on the assumption of constant curvature bending, the shape of the whole arm can be consequently estimated. As claimed by the author, this method applies to 2D position and shape sensing. Example (b) uses the trilateration method to locate the end-effector: four wire potentiometers are installed in fixed setpoints on an experimental jig, with the wire connected to the arm-tip; by measuring distances between the arm-tip and four setpoints, the location of the end-effector could be calculated via geometry.

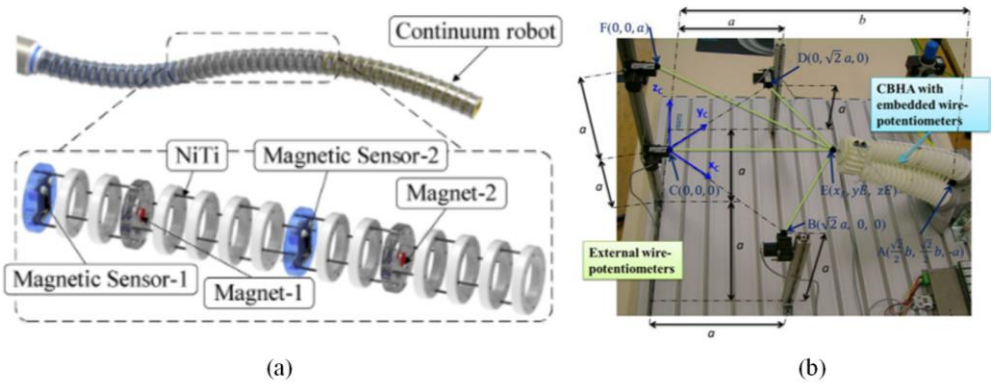


Figure 2-14 Examples of novel tip-position sensing methods for continuum robots that are not commonly used. (a) Shape estimation using permanent magnets and magnetic sensors [123]. (b) Triangulation system based on wire-potentiometers [58].

2.3.4 Summary of general sensing solutions

Figure 2-15 summarized the sensing method applied on continuum robots to measure the configuration or tip-position of the arm structure, which can be classified as contact or non-contact sensing solutions, the former includes self-sensing material, displacement sensors, optical sensors, etc. and the latter refers to external vision & imaging and EM tracker. Besides, several un-commonly used sensing methods are also introduced.

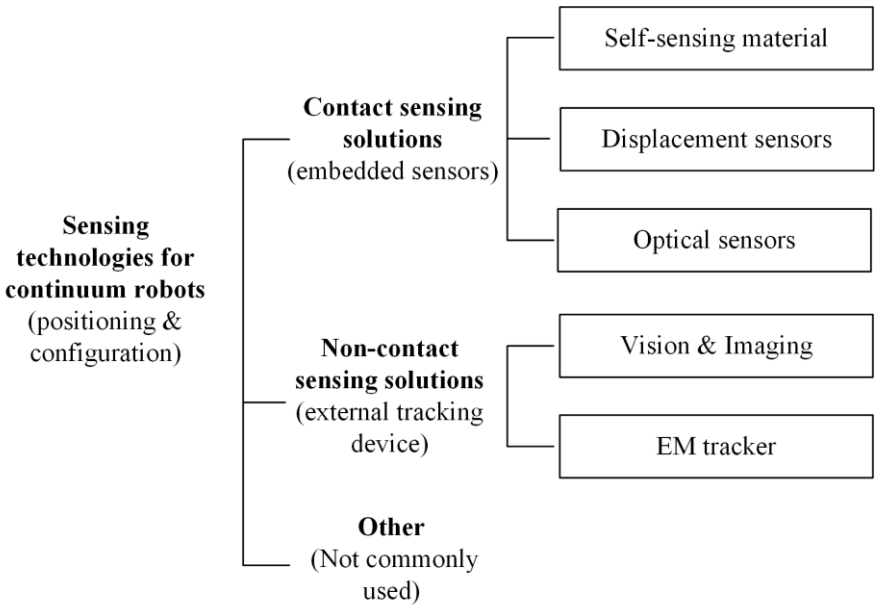


Figure 2-15 A summary of sensing technologies for continuum robots.

The non-contact sensing solutions relying on external devices are not applicable in required working conditions (i.e., in confined spaces such as in an aero-engine, channel, or nuclear box), while the contact sensing methods and vision-based devices show their potential. **In this research, a comprehensive sensing method will be explored and developed for a multiple-section cable-driven continuum robot to achieve closed-loop control.**

After reviewing various sensing solutions for shape reconstruction and tip-tracking of the continuum robot system, it was believed the best solution for our project may rely on developing an embedded vision system to equip the robot arm with a self-sensing ability (consequently to achieve closed-loop control in confined space). Thus, an additional survey for this type of sensing method not only for continuum robot systems but also for rigid-link robots and industrial robots, etc. is also conducted.

2.3.5 Embedded vision applications on continuum robot systems

For continuum robots, the most common case is the tip-mounted miniature (probe or endoscope) camera. Imaging is the basic function of this implement (Figure 2-16 (a) & (b)) to observe the working environment and enhance the view at the operating site. Figure 2-16 (a) demonstrates a slender continuum robot designed for engine inspection & in-situ repairing, two Naneye cameras are attached to the distal tip, providing images of the workspace, which allows the user to adjust the tip according to the surrounding environment and achieve interactive control. Figure 2-16 (b) introduces a vision-equipped fetoscope (i.e., the Naneye stereo cameras), in this application, the images returned by the camera not only provide surrounding information but are also processed by stereo reconstruction algorithm [124] to compute the geometry.

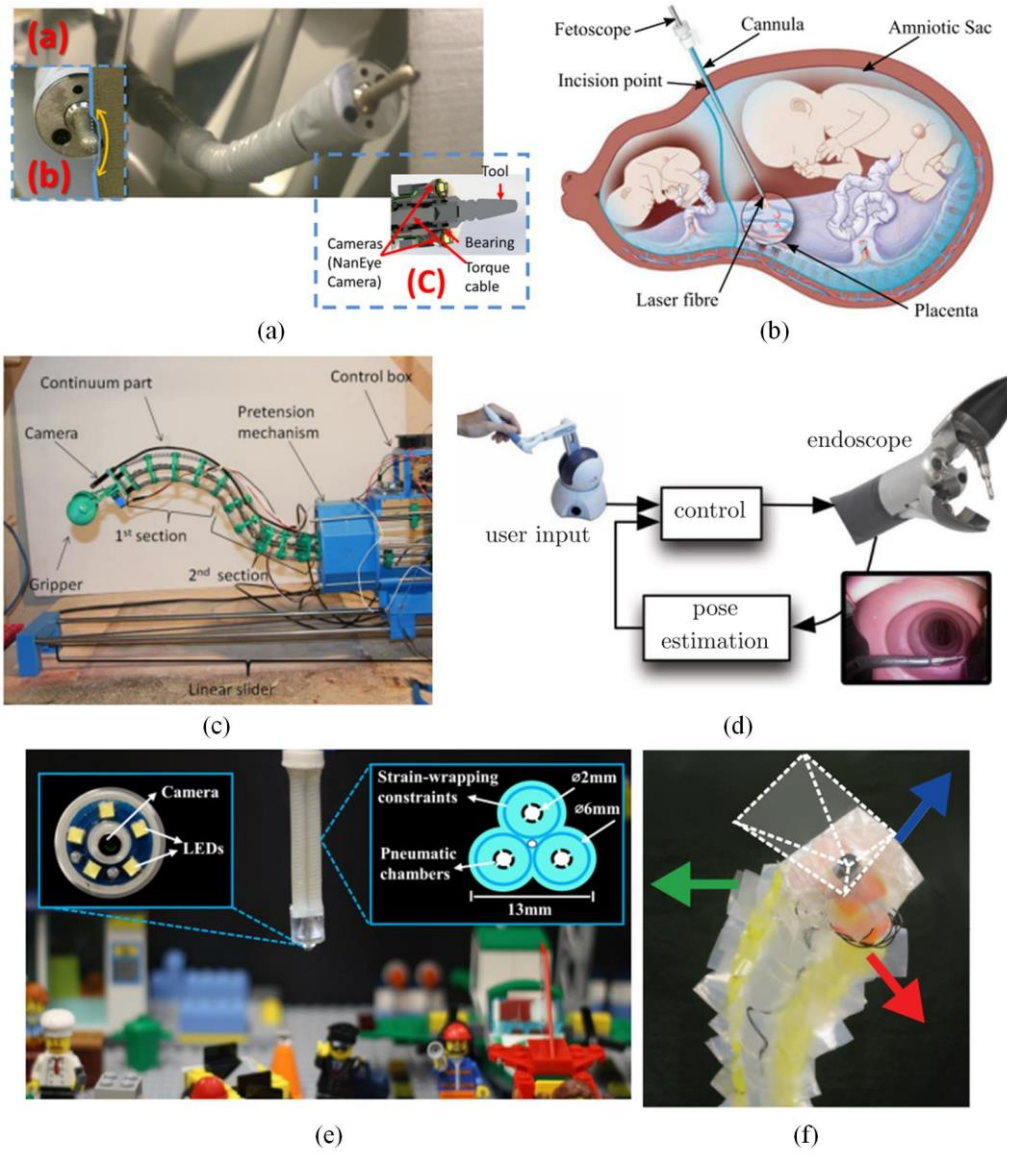


Figure 2-16 Applications of embedded visions on continuum robots for imaging and sensing. (a) A slender continuum robot designed with two monocle cameras embedded for in-situ inspection of air engines [9]. (b) A fetoscope equipped with tip cameras designed for placental scanning [125]. (c) A continuum robot arm and gripper for harvesting cherry tomatoes [126]. (d) Endoscopic images used to control the flexible instruments [127]. (e) An eye-in-hand application of a tip-attached camera on a continuum robot [128]. (f) Application of a tip-attached camera [129].

These cameras can also serve as a visual servoing (i.e., a visual servoing controls a robotic system by feedback information extracted from images [130]) to control the tip-attached tools as shown in Figure 2-16 (c) & (d). The former prototype is a farming continuum robot that employs the camera to detect matured tomatoes and measure the distance to guide the movement of the gripper. In the latter example, an endoscope with

a flexible tip-attached instrument designed for minimally invasive surgery is presented, and the pose of the flexible instrument is estimated by the endoscopic images during the operation. Another similar example can be found in [131], herein the feedback images are also used to investigate the non-linearities of the tip-attached flexible tool.

Further, the most advanced application is locating the robot arm itself through the images captured by tip-attached cameras (Figure 2-16 (e) & (f)). The eye-in-hand visual servo control law is applied in both examples (i.e., eye = camera, hand = robot). By comparing the feature points of the working environment recognized from a serial of images, the trajectory path or the absolute position of the robot tip can be detected. This method largely depends on the camera feedback rate, feedback quality, and the feature detection algorithm, thus, it is better to be applied in a structured working environment.

2.3.6 Embedded vision applications on other robotic systems

The earliest concept of vision-based control (visual servoing) is appeared in 1979 [132], then the related applications are widely used in conventional rigid-link robots. There are two basic camera-robot configurations in these applications: eye-in-hand and eye-to-hand, which are illustrated in Figure 2-17 (a) and (b). The eye-in-hand configuration describes a camera mounted on the end-effector of a robot while in the eye-to-hand configuration, the camera is fixed in the working environment to observe the robot from outside, both methods are applied based on hand-eye-calibration to figure out the position relation between the camera and the robot (the detailed process will be introduced later in Chapter 4). In comparison, the former configuration is manoeuvrable and has a precise sight of the scene [133] while the latter one has a wider view of the workspace [134]. A typical look-and-move servoing architecture is presented in Figure 2-17 (c), which may give a basic concept of the work frame of a visual servoing. In this case, the control target is the Cartesian motion of the end-effector, and the feedback images are processed to provide pose estimation of the robot, consequently, to calculate the Cartesian error.

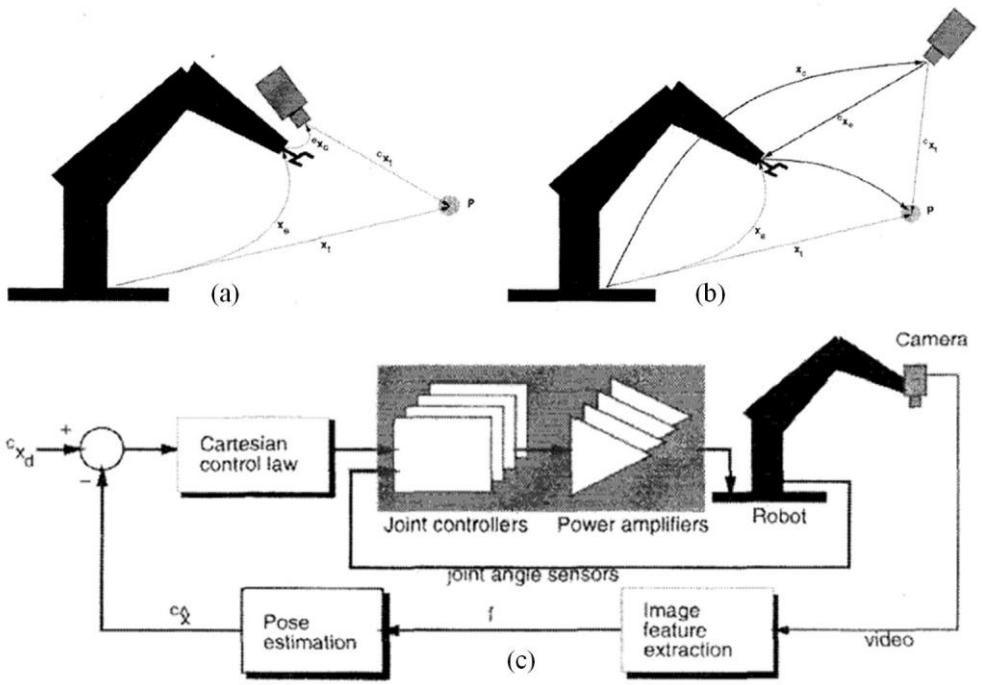


Figure 2-17 Basic conceptual framework of visual servoing (vision-based control) of robotic systems [135]. (a) The eye-in-hand configuration. (b) The eye-to-hand configuration. (c) A basic dynamic position-based look-and-move servoing architecture.

The later research tries to enhance the visual servoing control by using hybrid camera-robot configurations as shown in Figure 2-18. One of the cases utilizes two cameras to control one robot system (Figure 2-18 (a)), the static (global) camera is fixed to control the position of the end-effector via a mobile landmark attached to the robot, and mobile (tip-attached) one controls the orientation of the end-effector by observing the static target. In Figure 2-18 (b), two robots cooperate via one tip-attached camera, thus, the main robot is under eye-to-hand configuration and the secondary (guidance) robot is under eye-in-hand mode. Both methods increase the operatable workspace of the visual servoing as well as improve the control stability.

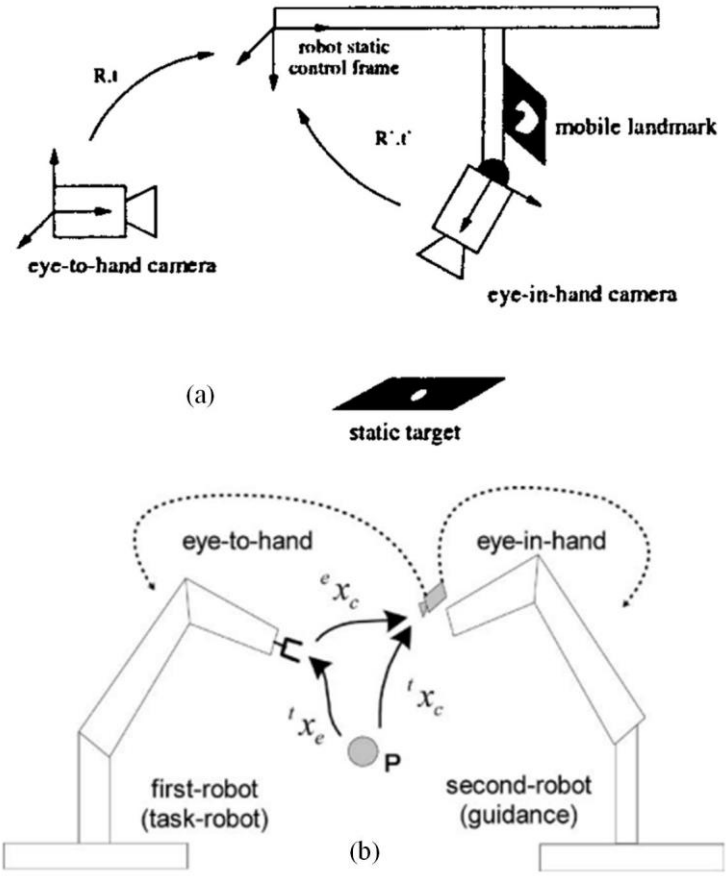


Figure 2-18 Combined configuration of a visual servoing. (a) Eye-in-hand / eye-to-hand cooperation for visual servoing [133]. (b) Eye-to-hand approach on eye-in-hand configuration [125].

Nowadays, the method of visual servo has been used in 3D global mapping (by integrating with SLAM) [136], vision-guided pick-and-place tasks [137], identifying the kinematic parameters of an industrial robot (with no prior information of the kinematics model) [138], etc. which also shows great potential in further task-oriented applications.

2.3.7 Summary of embedded vision applications

This section introduces the applications of embedded vision on both continuum robots and rigid-link robots. To sum up, on a continuum robot, the embedded cameras are mainly used to observe the external environment or attached tools, while they usually work as visual servoing (providing feedback for vision-based control) on industrial

robots. Recently, the concept of visual servoing is gradually employed in continuum robot systems with miniature cameras, which can be further explored and developed.

2.4 Controller designs for continuum robots

With the development of modeling and sensing devices, various intelligent controllers were developed for continuum robot systems. These controllers can be divided into two groups, model-based control, and model-less (model-free) control.

2.4.1 Model-based controllers

The conventional model-based controller depends highly on accurate modeling, either for static (kinematic) or dynamic control [19]. Nevertheless, in the open-loop control, the inherent characteristics of continuum robots make it challenging to establish a precision model to predict potential errors under variant working conditions, the low repeatability of the low-rigidity arm structure even leads to worse control performance. Recent research focuses on closed-loop control to solve these problems.

As the most widely used model, PCC is also commonly employed in model-based kinematic (static) control and shows good reliability and applicability. These controllers can be classified based on operating spaces [19, 53]. The joint and configuration space controllers usually rely on the sensors measuring the displacement of the actuation to eliminate the input control error (i.e., actuation cables, steel strings, etc.). But there are still unsolved problems regarding the cable elongation within the robot structure and the condition-dependent modeling error, bringing control uncertainty in configuration and task space. In comparison, the task-space closed-loop controllers work in positioning the robot tip in a 2-D or 3-D workspace directly, while the configuration-space controller controls the robot tip indirectly. Conventionally, inverse kinematics or dynamic models are established for a continuum robot to convert Cartesian tip movement to actuation tendon displacement.

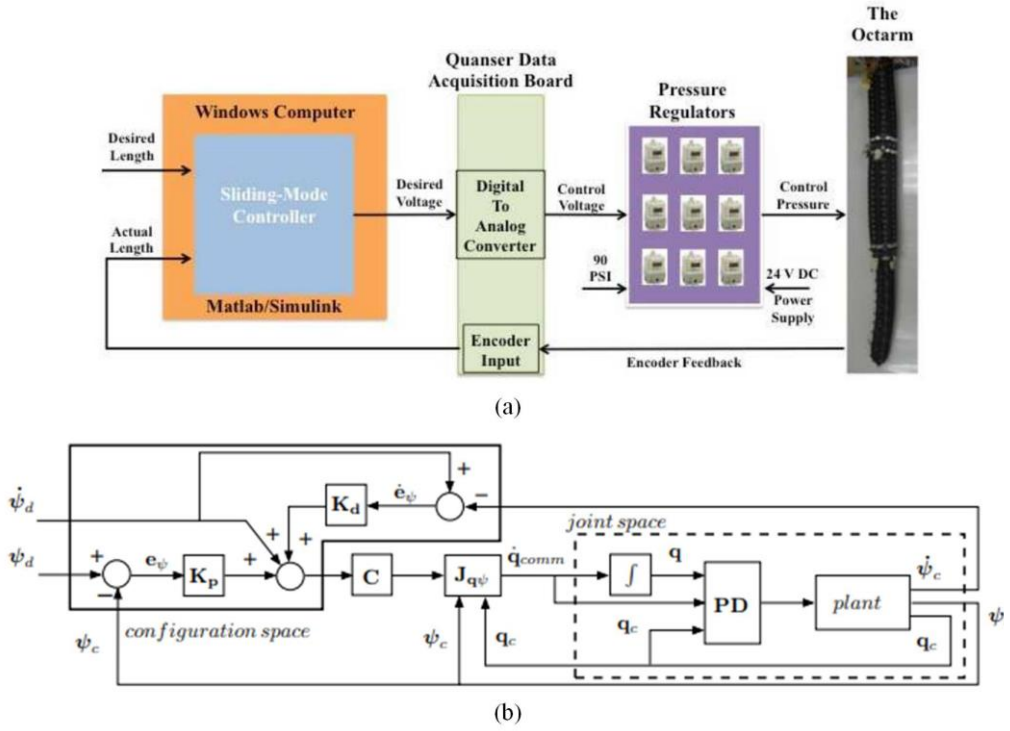


Figure 2-19 Architects of two configuration space controllers. (a) A sliding-mode configuration controller designed for Octarm (extensible pneumatic robot) for planner operation [53]. (b) A configuration controller designed for a cable-driven continuum robot [39].

The architectures of two model-based configuration space controllers are demonstrated in Figure 2-19. The controller (a) is designed for an extensible pneumatic continuum robot (the Octarm). Herein, the only feedback information is the tube lengths in joint space, and the control inputs and outputs are the pressure of pneumatic tubes and section lengths & curvatures of the arm in planner motion respectively. Figure 2-19 (b) illustrates a more complex control diagram of a multiple-section tendon-driven continuum robot introduced previously in Figure 2-13 (a). Here, both the extrinsic (i.e., tracking of tips by EM tracker as mentioned in Figure 2-13 (a)) and intrinsic (i.e., actuation cable lengths) sensory information is employed. The configuration space control target is the bending direction and bending angle of each section, thus, inverse kinematics is conducted based on the assumption of PCC to map from configuration angles to actuation cable lengths. The experiment results show a good performance of the controller in terms of decreasing coupling effects for multiple section control problems and increasing pose accuracy in bending & direction angle control.

Figure 2-20 compares a joint space controller (a) with a task space controller (b) designed for the same single-section catheter robot [19, 139]. Both the controllers employ tip position feedback via an EM tracker, in the tasks space controller, the positional error is directly sent to the controller while in configuration space control, it is converted to joint error first. In the path-following experiment (Figure 2-20 (c)), these two controllers demonstrate similar performance, and both of them have much better performance than open loop control.

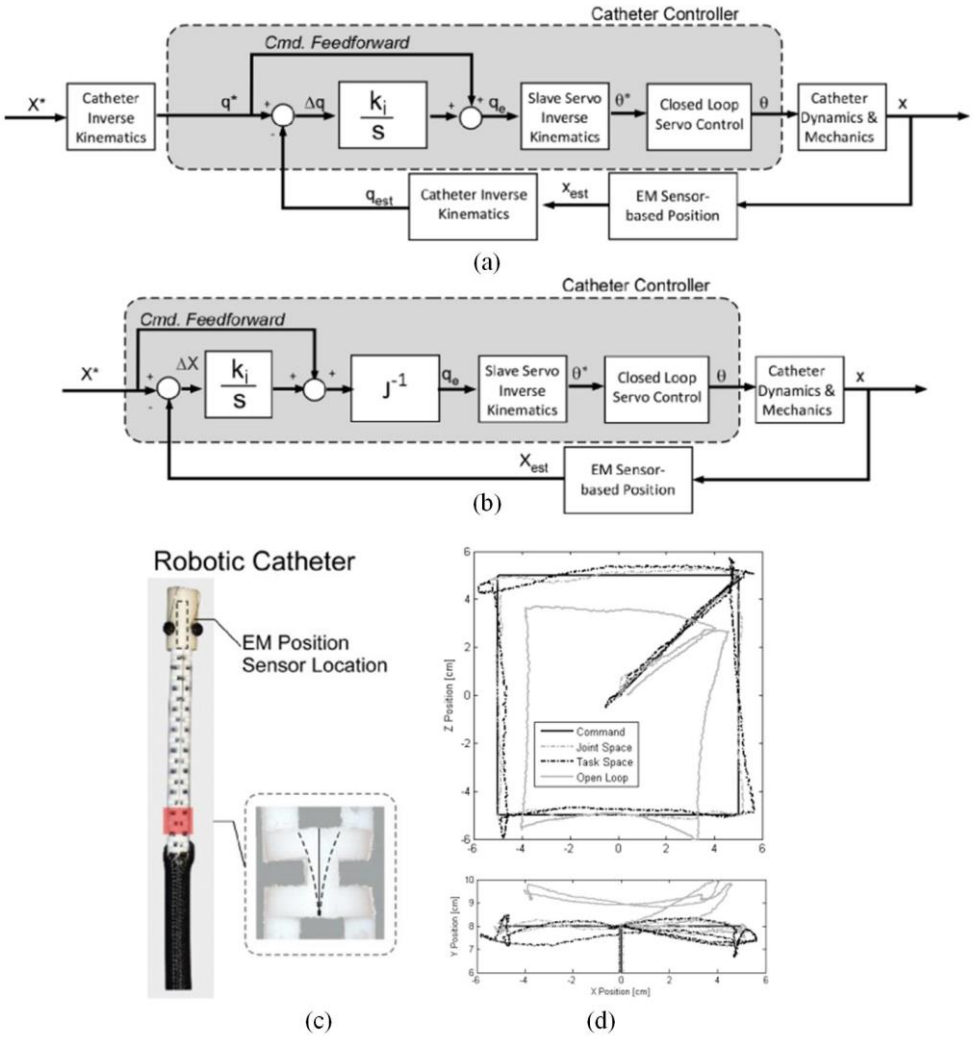


Figure 2-20 A comparison between the joint-space and task-space controllers [139]. (a) The joint-space control algorithm. (b) The task-space control algorithm. (c) The prototype robot. (d) The tip-tracking performance of the two controllers compared with open-loop control.

In conclusion, within the three types of model-based closed-loop controllers, regarding the positioning accuracy of the end-effector, a task space controller may theoretically

provide the best control accuracy if a reliable inverse kinematic model is available since it receives and deals with the tip position information directly. A joint space or configuration space controller depends more on precision forward kinematics to locate the end-effector (thus the performance is not guaranteed) while it provides a more stable and faster response.

2.4.2 Model-less controllers

In recent years, model-less or model-free methods have increasingly attracted attention to reduce the dependency on modeling and improve the adaptivity of a robot under different working scenarios. This type of controller usually utilizes a learning-based approach or empirical methods. For instance, the practice of online learning can help obtain an instant inverse mapping between the actuation and end-effector to predict the non-linear and time-varying model of continuum robots [98]. Another example is to build an empirical map from the task space to the actuation one based on the fuzzy logic algorithm to converge tip errors stably and efficiently in the closed-loop process [16].

In detail, one line of research focuses on supplementing the conventional kinematics or dynamics model of continuum robots with a learning-based prediction of uncertainty to compensate for control error. This method could be used either in position control, velocity control, or force control. In an early stage, a learning-based joint-space controller is developed on a 3-section cable-driven pneumatic extensible continuum robot [54], where the neural network was used in feed-forward control of cable-displacement to eliminate the uncertainty introduced by the actuation system. A similar method is applied to a bionic handling Arm for tip-position control [58], an feedbacked uncertainty modeling is added up to the feed-forward model during control, thus the controller is updated according to task space feedback (i.e., the tip position is obtained by trilateration method as introduced before in Figure 2-14 (b)) during the motion to achieve adaptive control, the path tracking experiment also validates this method (Figure 2-21 (d)). The model-less control could also be used in force-control. In a recent

research [140], a method using RNN to predict external force between the robot arm and working condition is proposed, aiming to achieve compliant control of the surgery robot. In detail, a central-backboned tendon-driven continuum robot equipped with motor encoders and force cells is used in this experiment (Figure 2-21 (a) & (b)), the RNN network maps the relationship between two sets of values: [cable displacement & force on each cable] and [external force applied]. After inserting this model into control loop, the robot will automatically adjust the cable force according to predicted external force, aim to reduce reaction force between robot and working condition thus achieve compliant control.

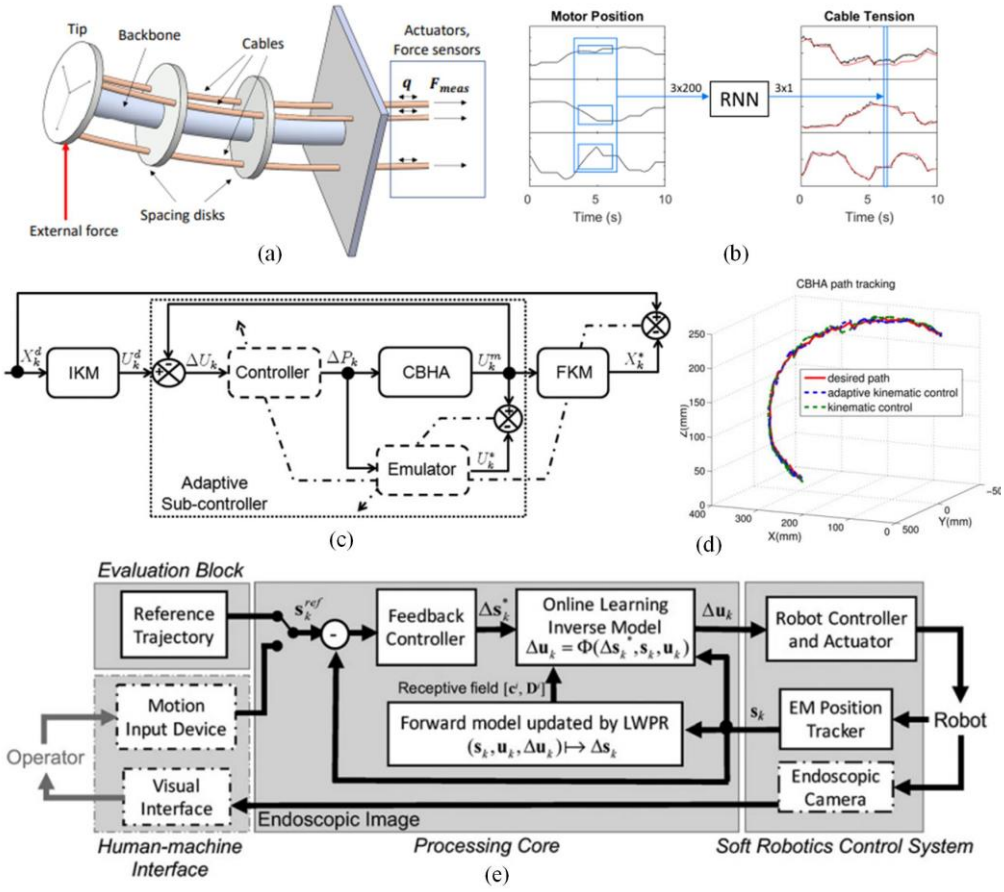


Figure 2-21 Examples of model-less & model-free control of continuum robots. (a) RNN-based compliant force control of a cable-driven continuum robot[140]. (b) RNN-based tension prediction [140]. (c) Adaptive neural network control for CBHA: the controller architecture [58]. (d) Adaptive neural network control for CBHA: the path tracking experiment [58]. (e) The control diagram of the LWPR learning-based control [98].

Another approach that completely drops the conventional kinematics & dynamics model in controller design, is called the model-free method. This method usually uses real-time collected data (such as tip-position, and actuation parameters like tendon length or pneumatic pressure) to map between control input and output, consequently achieving adaptive control. In [98], an inverse mapping obtained by online learning is integrated with a forward model based on the LWPR learning method to control a pneumatic driven continuum robot (Figure 2-13 (a) & Figure 2-21 (e)), which significantly reduces control error of tip-position. In his method the relationship between [last moment tip position & desired tip position] and [pneumatic pressure changes in each tendon] is mapped by locally weighted projection regression, these data update each time when the robot moves. The localization method also minimizes training time for each step to ensure real-time control. Later on, on the same robot, Ge F. (2019) did another trial that use a tip camera to feedback tip position against the environment in a training process and achieve adaptive control by a similar method [128].

To sum up, both the learning-based and empirical methods are seeking solutions to establish a feasible mapping between desired control input and output to achieve target control performance. The conventional learning methods have shown stable and feasible performance while it always has a time-consuming training process, and the idea of online learning can largely decrease the size training set. At the current stage, the optimal solution for continuum robot applications may integrate prior information referring to robot kinematics or dynamics with the learning-based approach to establish a hybrid control logic.

2.5 Navigation for continuum robots in the task space

In industrial practice, a continuum robot is always assigned for inspection, repair, and maintenance works in confined or narrow spaces, which places high demands on the navigation performance of such systems.

In academics, robot navigation can be defined as 1) the ability of a robot to locate itself in a reference or global coordinate (self-localization); 2) the ability to plan an optimal path from the start position to the target position and achieve safe navigation via obstacle avoidance (path planning); 3) some robots also have the ability to represent the working environment (map-building and map interpretation) [141-143].

2.5.1 Navigation technologies for conventional robot systems.

Nowadays, navigation technologies are widely applied to various robot systems such as mobile robots, unmanned air vehicles, industrial manipulators, etc., and diversiform sensors and methodologies are employed.

- Mobile robot (automatic vehicle) as an example

The navigation for mobile robots is the most studied topic. For localization, the inertial sensors [144], laser range finder [145], infrared sensors[146], and vision devices (camera)[147], are usually employed to determine the position and orientation of the robots. The vision-based sensors can also be used to extract environment features and rebuild (map) the working environment in 2-D or 3-D through the SLAM (Simultaneous Localization and Mapping) algorithm [148-150]. Path planning aims to output an optimal path between start and end points within a working scenario, that could be classified as off-line planning and on-line planning. The offline planning is also called global navigation, where the robot is supposed to know prior information of the working environment; while the online planning (local navigation) means the robot can update surrounding information instantly through embedded sensors [141, 151, 152]. Both the off-line and on-line path planning are essentially a search and optimization problem that can be solved via methods of artificial potential field (APF), A* algorithm, vector field histogram (VFM), particle swarm optimization, and so on [141].

- Industrial robots (manipulator robots)

Similar to continuum robots, industrial manipulators usually undertake assembly work, inspection, and some mechanical processes in industrial applications. Although this type of robot is always fixed at the base, the accuracy of motion trajectory control and navigation of the end-effector is still essential work.

In the early stage, the end-effector localization mainly depends on motor encoders (at the joints) and IMU (i.e., the inertial measurement unit), these sensors obtain the rotation angle and velocity of the robot joints or the measurement points on the robot, and then estimate the position or trajectory of the end-effector according to robot kinematics [153-155]. However, there is an inherent error introduced by deformation or mechanical imperfections of the manipulator, especially the accumulated errors for multiple link robots, and the accumulated error is usually difficult to formulate. To settle this problem, secondary sensors or multiple sensors are used, thus, multiple-source feedback can be combined and improve the localization accuracy. In most recent research, vision-based localization become more popular, which uses either eye-in-hand or eye-to-hand configuration (i.e., also refer to Chapter 2.3.6 and Figure 2-17) to determine the position of the end-effector by analyzing the feature points in the images [156, 157]. And similar to the mobile robot introduced in the above section, the vision devices mounted on the end-effector of an industrial robot also can map the environment in some cases [158].

The path planning problem for industrial manipulators is always an optimization process with some constraints such as workspace constraints, velocity, or acceleration constraints applied [159, 160], where the rapidly exploring random trees (RRT) [160, 161], probabilistic roadmap method (PRM) [162], artificial potential field [163] and fuzzy logic [164], etc. are some commonly used methods. From an initial point, these methods start a search by randomly generating the motion of the end-effector (by adjusting joint parameters), until a feasible solution is reached. Nowadays, the path

planning process and optimization can be computed under the CAD (computer-aided design) model environment[165, 166].

2.5.2 Navigation technologies for continuum robots

For continuum robot systems, the common shape sensing and tip localization methods have been discussed in Chapter 2.3 . However, because of the features of hyper-redundant structure and low sensing capability, there are limited cases regarding continuum robot path planning and navigation. Several researchers have advised the follow-the-tip motion for continuum robot path planning [167-170], which implements the methods of sequential quadratic programming optimization, improved RRT, and local search respectively.

Table 2 Robot Navigation Technologies

Sensors	*Localization*	*Path planning*	*Detection & Mapping*
Mobile robots			
• Vision sensors • Laser sensors • Encoders • Inertial sensors • Ultrasonic sensors • Multiple sensors fusion	• Triangular measurement method • GPS • Vision-based method • Land marker	• APF • A* algorithm • VFM • Particle swarm optimization	• SLAM • Ultrasonic
Industrial robots (manipulator robots)			
• Vision sensors • Encoders • IMU (Inertial)	• Photogrammetry • Model-based method	• RRT • PRM • Artificial Potential Field	• Vision-based method
Continuum robots			
Refer to Chapter 2.3		• Sequential quadratic programming optimization • Improved RRT • Local search	• Vision-based method

2.6 Opportunities for future research

Through a wide range of literature reviews presented in this chapter, the state-of-the-art development in the field of continuum robots was studied and summarized regarding mechanical design, modeling approach, sensing capability, control strategy, and application, etc. In summary, structural design, feedback systems, and controller design are three key elements contributing to the control accuracy of continuum robots. Since the main structure of our continuum robot system has been developed (although there are still some minor adjustments that can be made such as rearranging the cable distribution, reassembling the arm to adjust the section length, etc.), the following solutions could be considered to improve the controllability of the system:

- Need for a thorough error analysis for robot control

A thorough error analysis for current continuum robot systems from actuation space to task space is necessary, regarding cable displacement feedback and control, robot tip (s) tracking, and positioning. By evaluating these error sources, potential improvements and solutions can be proposed to minimize the control error of a robot.

- Need for comprehensive sensing solutions

To achieve precision control of a continuum robot in a confined space, a compactly designed sensing system is required. The sensing solution could be considered in terms of operation spaces. To be specific, in joint space control, the sensing object is the displacement or elongation of actuation cables, rods, or tubes, where the displacement sensors are applicable as mentioned in Chapter 2.3.1 . In configuration space, the direction and curvature of the manipulators (robot arm) need to be figured out, especially in navigation or tip-following tasks where the body pose is an important consideration to evaluate the robot performance while finding a direct tip-tracking method perhaps the best solution in task space. Excepting select optimal sensing devices

or solutions for each operation space, a hybrid sensing solution could also be taken into account to enhance the sensing ability.

- Need for a closed-loop controller cooperating with the sensing system for multiple-section control

The controller design is another concern followed by proposing a proper sensing solution. The control accuracy, stability, and feasibility should be considered when designing the control framework based on the available sensor signals.. For a single section robot, the solutions can be sought from closed-loop model-free or model-less control based on the available sensing signal, minimizing the control error introduced by theoretical modeling. Hybrid control (i.e., combing model-based control with model-free/ model-less control) could also be investigated. For a multiple section robot, the model-free methods may be not applicable since they always rely on training, and the training sets will increase exponentially as the number of sections increases.

After achieving accurate control through proper sensing and control algorithms, more applications and working scenarios of a continuum robot can also be explored such as navigation and inspection in pipe work, for nuclear facilities, etc.

- Need for broadening working scenarios for a continuum robot

The practicability and potential working scenarios of a continuum robot need to be further explored and developed. For example, proposing an efficient navigation strategy can help the continuum robot go through a constructed space easily; and a feasible wireless control approach allows continuum robots to work in hazardous environments with safe operation.

Chapter 3 Basic Modeling, Formulation, and Control Challenges for the Prototype Robot

As introduced in Chapter 1 , the prototype robot is a two-section cable-driven continuum robot with a rotary actuation design. This chapter gives an introduction to the basic forward & inverse kinematics model implemented in the following research work. Then, the control challenges of this specific robot are discussed to identify the problems and possible solutions.

3.1 Basic concepts in continuum robot control

As mentioned in Chapter 2 , there are four operation spaces that refer to continuum robot control: actuation space, joint space, configuration space, and task space (workspace).

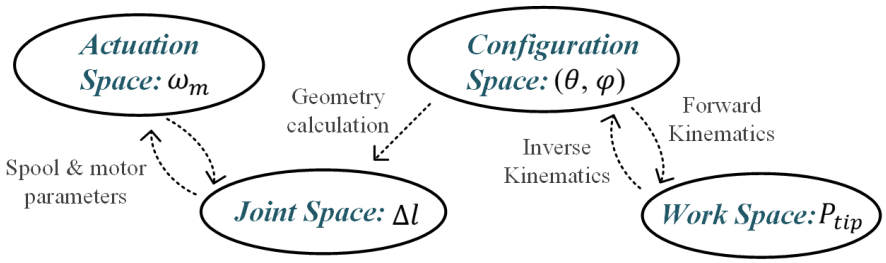


Figure 3-1 The operating spaces of the prototype continuum robot with corresponding control variables.

For this prototype robot, as shown in Figure 3-1, the actuation space and joint space control refer to the actuation motor and actuation cables respectively. The configuration space indicates the robot arm arc parameters as bending angle (θ) and direction angle (φ) of each section (Figure 3-2), the former represents the degree of the arc, and the latter indicates the rotation of the bending plane related to reference coordinate at the robot base. The task space refers to position (P_{tip}) control of the distal tip in the world coordinate. In this study, the primary control input of the robot controller is the configuration parameters (θ, φ), thus, the tip position P_{tip} can be calculated by forward kinematics. Inversely, if the desired tip position is known in a task-oriented work, the

robot poses of (θ, φ) can be deduced via inverse kinematics. After determining the value of (θ, φ), the cable displacement will be figured out by a series of geometry calculation (will be presented later in part 3.3), and the actuation cables will be driven by the actuation motors to adjust the pose of the robot.

3.2 Kinematics between configuration space to task-space

3.2.1 The PCC forward kinematics

As discussed in Chapter 2.4 , the controller for continuum robots can be mainly classified into two categories: model-based controller and model-less or model-free controller. The formers are usually open-loop controllers which are dependent greatly on precise modeling with time-consuming calculations. By contrast, the model-less or model-free controllers always try to simplify the model and reduce the calculation time, integrating with a closed-loop control system, this kind of controller can converge the error within the desired range [16, 98].

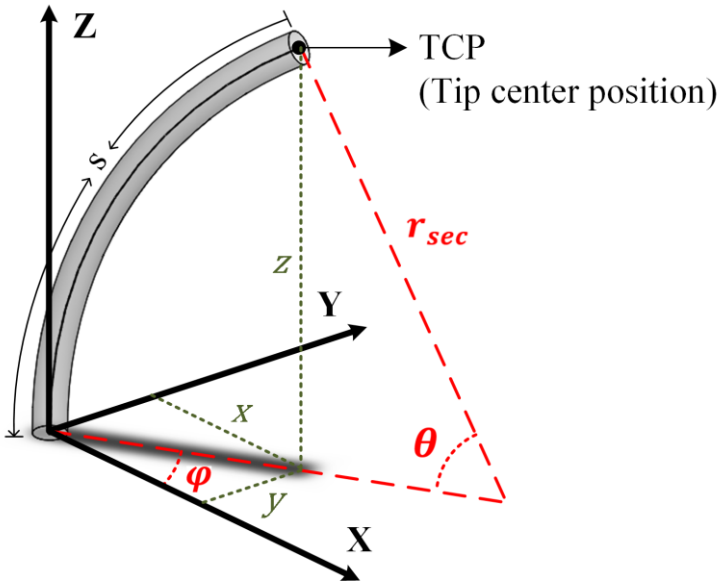


Figure 3-2 The PCC (piecewise constant curvature) assumption of a single-section continuum robot.

In this research work, a multiple-sensor integrated robot controller is developed to improve the control accuracy of a two-section continuum robot (will be demonstrated in Chapter 4), which employs a model-less closed-loop control algorithm design. To

simplify the calculation process and minimize the computing time, the general PCC (piecewise constant curvature) forward & inverse kinematics model is applied. Generally, the PCC assumption is a universal method that simplifies each section of a robot as a constant curvature arc, regardless of the detailed arm or actuation design. Basically, the PCC model introduces two joint angles (configuration parameters) θ and φ to describe the robot pose (shown in Figure 3-2) and treat them as primary control values. For the prototype robot, the bending angle θ is ranged in $[0, \frac{\pi}{2}]$, and the direction angle is ranged in $[0, 2\pi]$.

For **a single-section continuum robot**, the pose of TCP (i.e., tip center position, as shown in Figure 3-2) can be described by position P_{TCP} and orientation R_{TCP}, which can be expressed by the following equations:

$$r_{sec} = \frac{s}{\theta} \tag{3}$$

(i.e., s is the length of a single section, r_{sec} is the radius of the arc.)

$$P_{TCP} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r_{sec} \times (1 - \cos\theta) \times \cos\varphi \\ r_{sec} \times (1 - \cos\theta) \times \sin\varphi \\ r_{sec} \times \sin\theta \end{bmatrix} \tag{4}$$

$$R_{TCP} = Rot_z(\varphi) \times Rot_y(\theta) \times Rot_z(-\varphi) \tag{5}$$

$$= \begin{bmatrix} \cos\varphi & -sin\varphi & 0 \\ sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \times \begin{bmatrix} \cos -\varphi & -\sin -\varphi & 0 \\ \sin -\varphi & \cos -\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos^2\varphi\cos\theta - \cos^2\varphi + 1 & \cos\varphi\sin\varphi(\cos\theta - 1) & \cos\varphi\sin\theta \\ \cos\varphi\sin\varphi(\cos\theta - 1) & \cos^2\varphi + \cos\theta\sin^2\varphi & \sin\varphi\sin\theta \\ -\cos\varphi\sin\theta & -\sin\varphi\sin\theta & \cos\theta \end{bmatrix}$$

(i.e., $Rot_K(\emptyset)$ denotes to rotation operator which performs a rotation about the axis $\widehat{K}$ of a reference coordinate by $\emptyset$ degrees).

Thus, the base coordinate and tip coordinate of a single-section robot can be linked by a transformation matrix ${}^{base}_{tip}T$:

$$ {}^{base}_{tip}T = \begin{bmatrix} R_{TCP} & P_{TCP} \\ 0 \;\; 0 \;\; 0 & 1 \end{bmatrix} \tag{6} $$

In the case of **multiple-section robots**, the PCC assumption can be treated as a re-entrant function, where n is the number of sections of this robot:

$$ {}^{base}_{n}R = {}^{base}_{n-1}R \; {}^{n-1}_{n}R \tag{7} $$

$$ {}^{base}_{n}P = {}^{base}_{n-1}R \; {}^{n-1}_{n}P + {}^{base}_{n-1}P \tag{8} $$

And the transformation matrix can be written as:

$$ {}^{base}_{n}T = {}^{base}_{1}T \; {}^{1}_{2}T \; {}^{2}_{3}T \ldots {}^{n-1}_{n}T = \begin{bmatrix} {}^{base}_{n}R & {}^{base}_{n}P \\ 0 \;\; 0 \;\; 0 & 1 \end{bmatrix} \tag{9} $$

Inversely, the base coordinate can also be described in the tip coordinate:

$$ {}^{n}_{base}T = {}^{base}_{n}T^{-1} \tag{10} $$

Figure 3-3 gives an example of a 3-section continuum robot ($n = 3$), with its base and section coordinates:

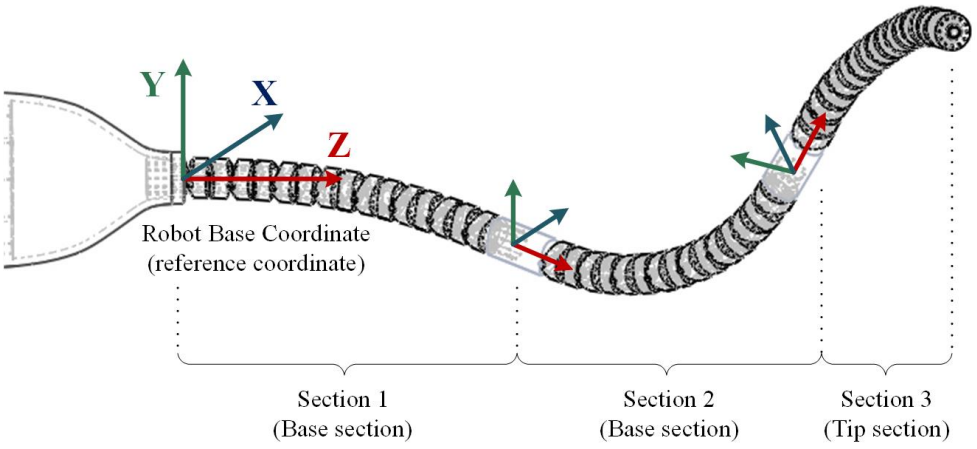


Figure 3-3 An example of a 3-section continuum robot, whose shape curves with continuous tangent vectors.

3.2.2 The modified PCC-based single-variable inverse kinematics

In task-oriented works, inverse kinematics is used more frequently to transform the cartesian space of the robot tip position $P_{tcp} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$ to the configuration space (i.e., robot pose or arc parameters of (θ, φ)) or joint space (i.e., cable displacement). This is often computationally expensive and is not often solvable by numerical methods, since it is always a multi-solution problem with non-linearities.

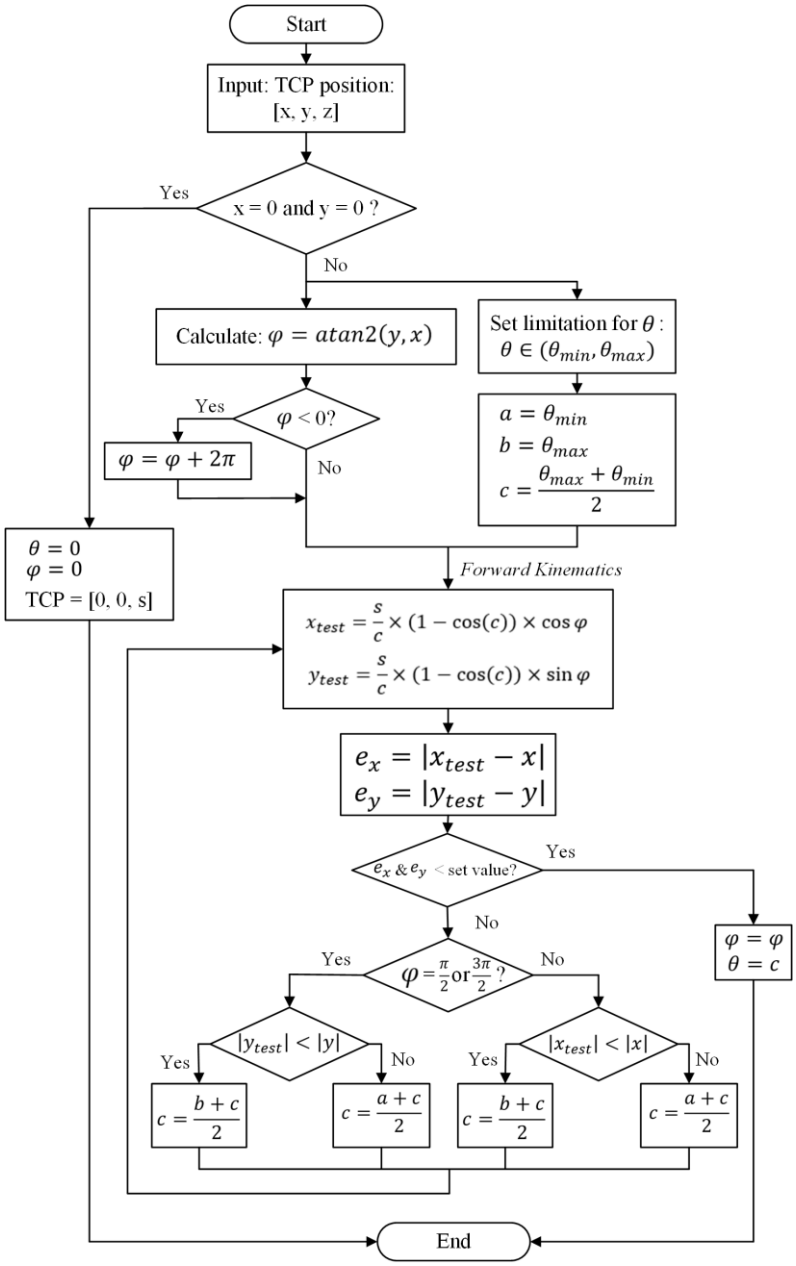


Figure 3-4 Inverse kinematics for a single-section robot (based on bisection method).

Thus, for inverse calculation, a simplified iterative method is proposed (Figure 3-4). The input is the desired TCP position, and the output is the corresponding robot configuration $[(\theta, \varphi)]_n$.

For a single section robot with a known TCP position $[x, y, z]'$, the direction angle φ is a fixed value that can be directly calculated by $\varphi = atan2(y, x)$, as shown in Figure 3-2. Then, the bending angle θ ranged within $(\theta_{min}, \theta_{max})$ can be computed by analytical methods such as the bisection method (i.e., θ_{min} and θ_{max} can be the defined limitation of the robot bending angle or an estimated value). The bisection method figures out the value of θ by repeatedly bisecting the interval between $(\theta_{min}, \theta_{max})$, and then selecting the subinterval where the decision equation(s) changes sign until the desired error tolerance is reached. The precondition is that the tip position $[x, y, z]'$ must be in the workspace of the robot arm. In this case, **the arc parameters of the robot are decoupled, and the only variable in the numerical process is the bending angle of the robot rather than a pair of $(\boldsymbol{\theta}, \boldsymbol{\varphi})$.** Besides, there is also no complex matrix operation applied, which significantly simplifies the inverse kinematics regarding the formulation and computing time.

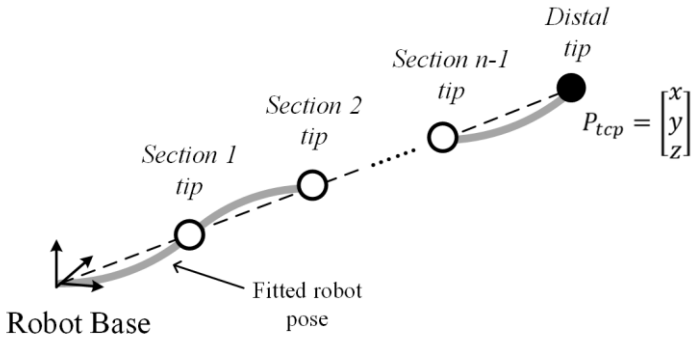


Figure 3-5 Sketch for a multiple-section problem.

For a multiple-section problem (Figure 3-6), the desired configuration is computed section-by-section. Firstly, the desired Tip position of the distal tip $[x, y, z]'$ is divided into n subsections, representing the section tips as listed in equation (11) and Figure 3-5.

$$P_1 = \frac{1}{n}\begin{bmatrix} x \\ y \\ z \end{bmatrix};\ P_2 = \frac{2}{n}\begin{bmatrix} x \\ y \\ z \end{bmatrix};\ \dots;\ P_n = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{11}$$

Then, the method follows the rules developed on the single-section inverse kinematics presented in Figure 3-4, which is applied to fit the robot pose for each section sequentially. In this case, a later section i is located in the coordinate of the prior section(s) ${}_{i-1}^{0}T$ to find the best-fitting configuration (Figure 3-6).

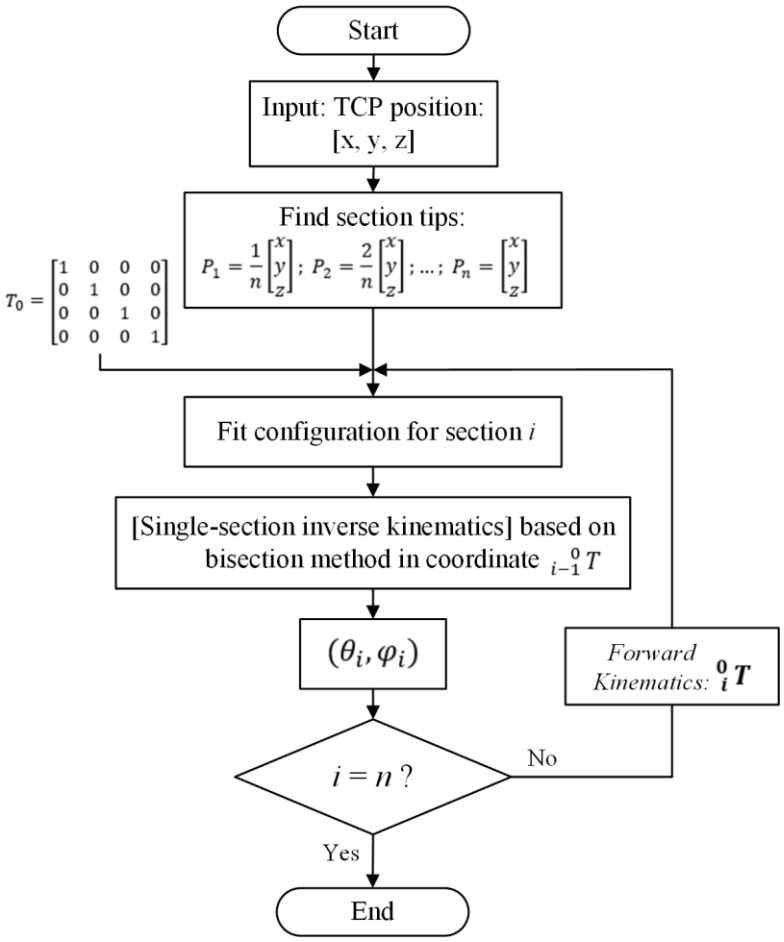


Figure 3-6 Workflow chart of the multi-section inverse kinematics.

3.3 Kinematics from configuration space to joint space

To calculate the desired cable displacement needed to achieve the desired configuration (θ, φ) of a section, a set of equations can be derived from the method proposed by Jones et. al [27]. Jones's paper describes the method to calculate the cable length of a 3-cable driven section whose cables are spaced equally, while the actuation cables of the prototype robot are distributed unevenly (Figure 3-7).

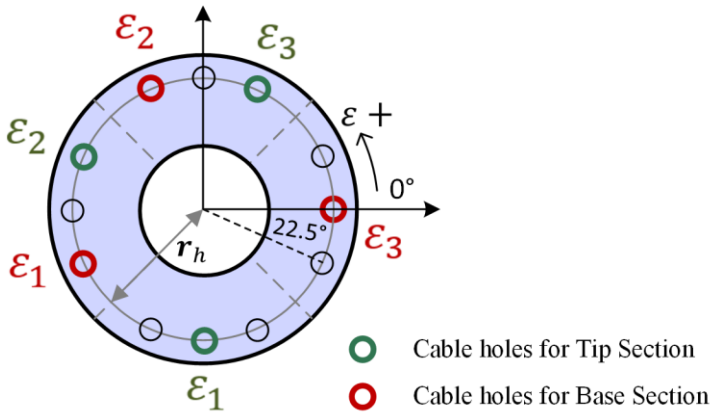


Figure 3-7 Cable hole distribution through the disks of the prototype robot, r_h is the radius of the cable hole pattern.

As explained above, in the PCC assumption, each section of the robot is modeled as a constant curvature curve, however, the actuation cables passing through the disks will take the shortest route between adjacent disks. Thus, according to Figure 3-8, the cables can be represented as a series of lines with equal length and the angle between adjacent lines are equivalent.

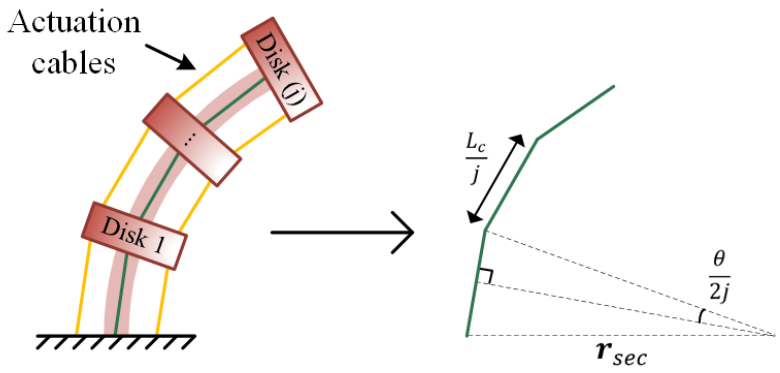


Figure 3-8 Graphical representation of actuation cables passing through the disks.

Assuming there is a virtual cable passing through the center of the disks (L_c) Figure 3-8, the cable length between adjacent disks can be formulated as:

$$\frac{L_c}{j} = 2 \times \sin(\frac{\theta}{2j}) \times r_{sec} \tag{12}$$

Where j is the number of disks in this section; θ is the bending angle of this section; r_{sec} is the radius of the section arc.

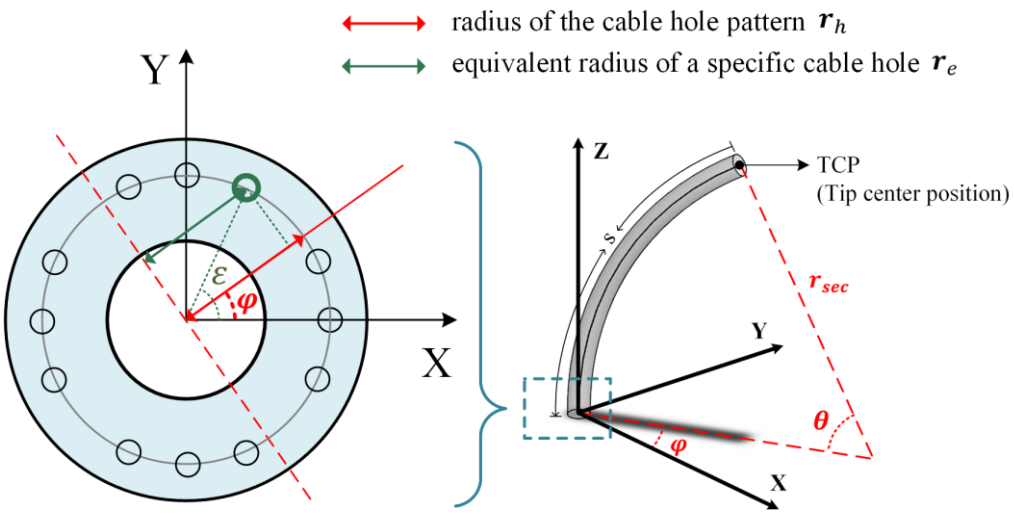


Figure 3-9 Illustration for the equivalent radius of a specific cable hole when the bending angle is φ.

Then the concept of equivalent radiuses r_e (Figure 3-9) are introduced to calculate the cable length for a specific cable whose guiding hole is allocated at the angle of ε:

$$r_e = r_h \times |\cos(\varepsilon - \varphi)| \tag{13}$$

Thus, the general cable length L_d at angle ε (i.e., cable hole location) can be calculated as:

$$\frac{L_d}{L_c} = \frac{r_e}{r_h} \tag{14}$$

$$L_d = 2j \times \sin(\frac{\theta}{2j}) \times r_{sec} \times |\cos(\varepsilon - \varphi)| \tag{15}$$

For a multiple-section problem, since the actuation cables of a distal section inevitably pass through a proximal one, thus the cable lengths of the distal section are the sum of the sections before:

$$L_d = \sum_{i=0}^{n} (L_{di}) \tag{16}$$

The cable hole distribution angles of the prototype 2-section robot are listed in Figure 3-7 and Table 3:

Table 3 Cable hole distribution angles for both sections.

	Base Section	Tip Section
ε_1	202.5°	270°
ε_2	112.5°	157.5°
ε_3	0°	67.5°

For the base section (Section 1):

$$L_{d1-1} = 2j \times \sin(\frac{\theta_1}{2j}) \times \frac{s}{\theta_1} \times \left|\cos(202.5° - \varphi_1)\right| \tag{17}$$

$$L_{d1-2} = 2j \times \sin(\frac{\theta_1}{2j}) \times \frac{s}{\theta_1} \times \left|\cos(112.5° - \varphi_1)\right| \tag{18}$$

$$L_{d1-3} = 2j \times \sin(\frac{\theta_1}{2j}) \times \frac{s}{\theta_1} \times \left|\cos(-\varphi_1)\right| \tag{19}$$

For the tip section (Section 2):

$$L_{d2-1} = 2j \times (\sin\left(\frac{\theta_1}{2j}\right) \times \frac{s}{\theta_1} \times \left|\cos\left(270° - \varphi_1\right)\right| + \sin\left(\frac{\theta_2}{2j}\right) \times \frac{s}{\theta_2} \times \left|\cos\left(270° - \varphi_2\right)\right|) \tag{20}$$

$$L_{d2-2} = 2j \times (\sin\left(\frac{\theta_1}{2j}\right) \times \frac{s}{\theta_1} \times \left|\cos\left(157.5° - \varphi_1\right)\right| + \sin\left(\frac{\theta_2}{2j}\right) \times \frac{s}{\theta_2} \times \left|\cos\left(157.5° - \varphi_2\right)\right|) \tag{21}$$

$$L_{d2-3} = 2j \times (\sin\left(\frac{\theta_1}{2j}\right) \times \frac{s}{\theta_1} \times \left|\cos\left(67.5° - \varphi_1\right)\right| + \sin\left(\frac{\theta_2}{2j}\right) \times \frac{s}{\theta_2} \times \left|\cos\left(67.5° - \varphi_2\right)\right|) \tag{22}$$

Where (θ_1, φ_1) and (θ_2, φ_2) are the configurations for Section 1 and Section 2 respectively.

3.4 Modeling and control challenges for the prototype robot system

As introduced in Chapter 2 , in the field of continuum robotics, there are two main approaches for driving actuation cables, which use linear or rotary actuation units. The prototype robot employs a rotary actuation design (i.e., the motor-spool structure, as

demonstrated in Figure 1-4) since compared with linear actuators, the rotary ones can remarkably reduce the volume of the actuation system [11, 17]. However, it challenges the control and modeling of the continuum robots.

Cable displacement control accuracy is a primary objective for controlling continuum robots. Generally, a spool system is enrolled to connect to the rotary motor with a cable attached and coiled on it for a rotary actuation unit (Figure 1-4). There are two primary error sources for the cable-displacement estimation: (1) the incorrect conversion between the motor rotation and cable linear movement caused by the uncertain practical cable windings (Figure 3-10 (a-c)); (2) the cable elongation raised by the applied force along it. In an ideal situation, the cable is fully tensioned and winding on the spool with a constant pitch(p) and diameter(d) for each revolution (Figure 1-4). The relation between the motor rotation and the cable displacement is defined in Equation (1).

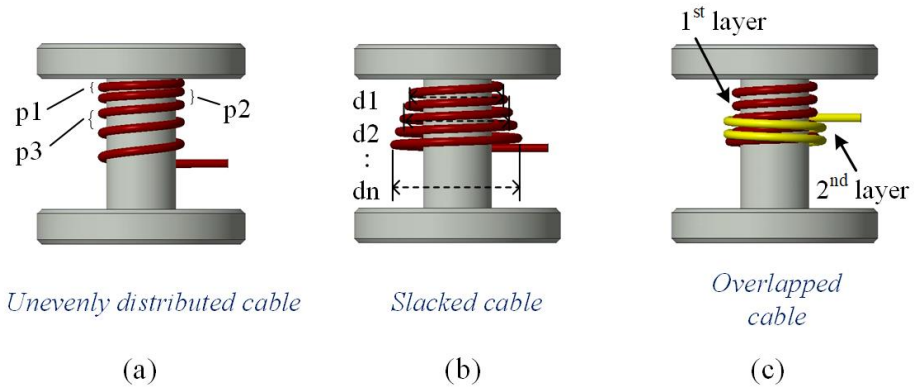


Figure 3-10 Cases of unideal winding of cables on the spools.

This calculation can theoretically predict the cable displacement based on the motor rotation measured by the rotary encoder, provided that the cable is properly tensioned, and the pitch is constant between the adjacent revolutions. However, several practical issues can happen in the spool system, such as uneven cable distribution, cable slacking, overlapping, etc. (Figure 3-10 (a-c)). These issues cannot be easily monitored by the existing sensing approaches, making the theoretical calculation invalid and bringing a

significant challenge to the modeling of the cable displacement, and affecting the positional accuracy of a continuum robot.

Further, as another challenge, cable elongation exists along the entire length of the cable in both the actuation system and the continuum arm, which can primarily affect the prediction of the cable displacement and the control of the robot. The inhomogeneous time-varying friction distribution along the cable even makes it harder to model the elongation. The friction distribution is affected by various factors, such as the shape of the arm, initial cable tension, and the cable channel material [171]. The impact of the cable elongation on the accuracy of the robot control is non-negligible, especially once the cable length is considerably large (>500mm) and the arm diameter is small. According to PCC (piecewise constant curvature) kinematics model, a small amount of cable-displacement difference will result in a large bending-angle difference of the whole robot. Besides, the friction along the driving cables also introduces time delay and backlash to the robot control, which slows down the responding speed and decreases the repeatability of the system. Hence, a sensing approach is needed for slender continuum robots to measure and monitor the cable elongation in real time for accurate positional controls of the shape of the arm.

In practical work, the multiple-section design even amplifies the effects introduced by the problems discussed above, since the actuation cables of a distal section inevitably pass through a proximal one, these cables always can affect the shape of the proximal section once the tension force varies. This will lead to coupled motions between sections, which challenges the model-based approach of continuum robot control.

Overall, as a trade-off of using the space-saving spool mechanism (rotary actuation), predicting the actual cable lengths is one of the main challenges, which also needs to consider the cable elongation caused by the friction along the cable and the uncertain cable windings. The multiple-section design also introduces the coupling problem, compounding the control challenges.

Besides the problems existing in mechanical structure design, another challenge is raised from the control strategy. Conventionally, kinematics model-based controllers (for instance, the PCC-based controller) and their derivative methods are widely enrolled in the controller design of continuum robots. However, these models are usually established in ideal circumstances with no consideration of self-weight, end payload, etc., and the deviations relative to structure design are unpredictable in the models. Thus, it is difficult to compensate for these inconsistencies (i.e., errors between theoretical prediction and practical operation) in controller design. Even though some recent research introduces various types of integral method to analyze these external influence factor, there is a lack of studies in multiple-section problems, and the accuracy of the controller with the computing-resource-consuming process is still limited, which introduce challenges in practice.

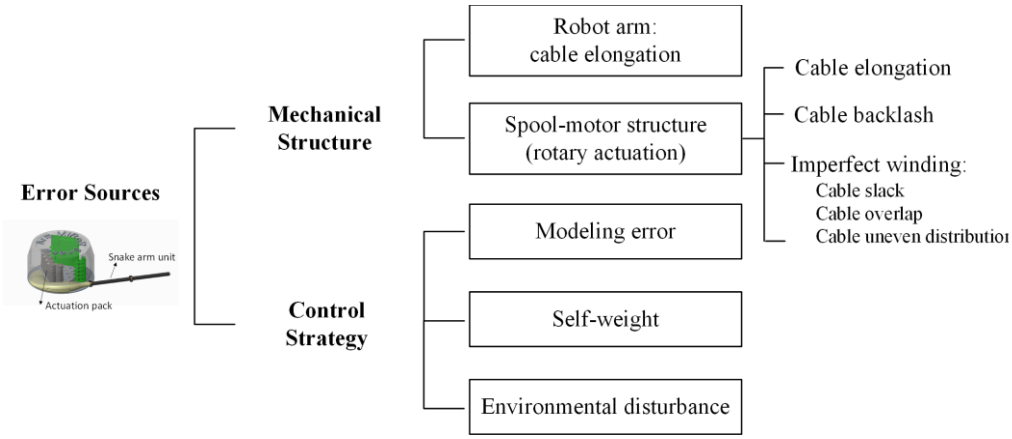


Figure 3-11 Error analysis for the prototype continuum robot.

To sum up (Figure 3-11), the challenges existing in continuum robots could mainly be concluded to be inexact cable displacement control resulting from elongation, the rotary actuation design, and the unrealized theoretical model used in the control algorithm. Since the difficulties in the quantities study of these problems, to achieve desired control accuracy both in joint space and work space, it is necessary to develop a comprehensive sensing system to provide feedback on the actual cable displacement and real-time tip position.

To deal with the problems listed above, a new design concept of a hybrid sensing system is proposed in Chapter 4 , which consists of a dual displacement sensor system implanted at the proximal end of the arm structure and a self-carried vision system to provide section-by-section tip-tracking. Further, a two-level closed-loop control approach is proposed for the multiple-section continuum robot, aiming to increase the tip positioning accuracy.

3.5 Chapter summary

This chapter introduces the basic modeling and formulation for the prototype robot applied in this study, which also proposes a modified PCC-based single-variable inverse kinematics, providing a simplified numerical method to solve the inverse kinematics problems. Then, a discussion of the control challenge of the specific two-section cable-driven continuum robot with a motor-spool actuation mechanism is presented. The purpose is to give the readers an overall understanding of our robot system and make it easier to present the following chapters about sensing system integration and controller design.

Chapter 4 A Hybrid Sensing System Integrated Closed-loop Controller Designed for a Multiple-section Continuum Robot

As summarized in Chapter 2, feedback systems and controller design are two key elements contributing to the control accuracy of continuum robots. This chapter proposes a hybrid sensing system (Figure 4-1) that is able not only to compensate for the tendons' displacement uncertainties (e.g., elongation, positioning on the spool winding system) but also minimize the coupling effect between the sections of the continuum robot (a major problem in multi-segmented snake robots).

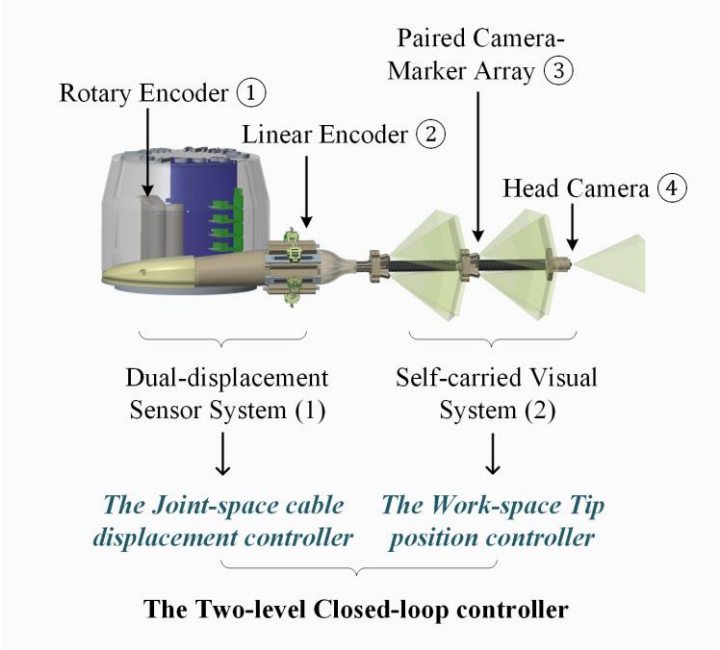


Figure 4-1 An overview of the hybrid sensing system and the cooperative two-level closed-loop controller.

This sensing system is distributed along the continuum arm: (i) for the coupling effect, a self-carried visual tracking system formed by several paired observe-reference camera & marker arrays are mounted at the base of each section to correct the pose of the arm; (ii) to improve the cable displacement control accuracy, a linear-encoder system is embedded to measure the displacement error within actuation pack and predict the elongation in the arm structure. Equipped with such a comprehensive sensing system

and a novel closed-loop controller (i.e., a two-stage cable-displacement control approach employing the linear encoder feedback is proposed for joint-space cable displacement control; and an indirect-measurement-based tip adjustment strategy integrating the tip-tracking feedback from the self-carried visual tracking system is devised for work-space control, as shown in Figure 4-1), the continuum robot (two-section) can reduce the distal tip control error by 73.5% compared with conventional methods, for both smooth and sharp movements.

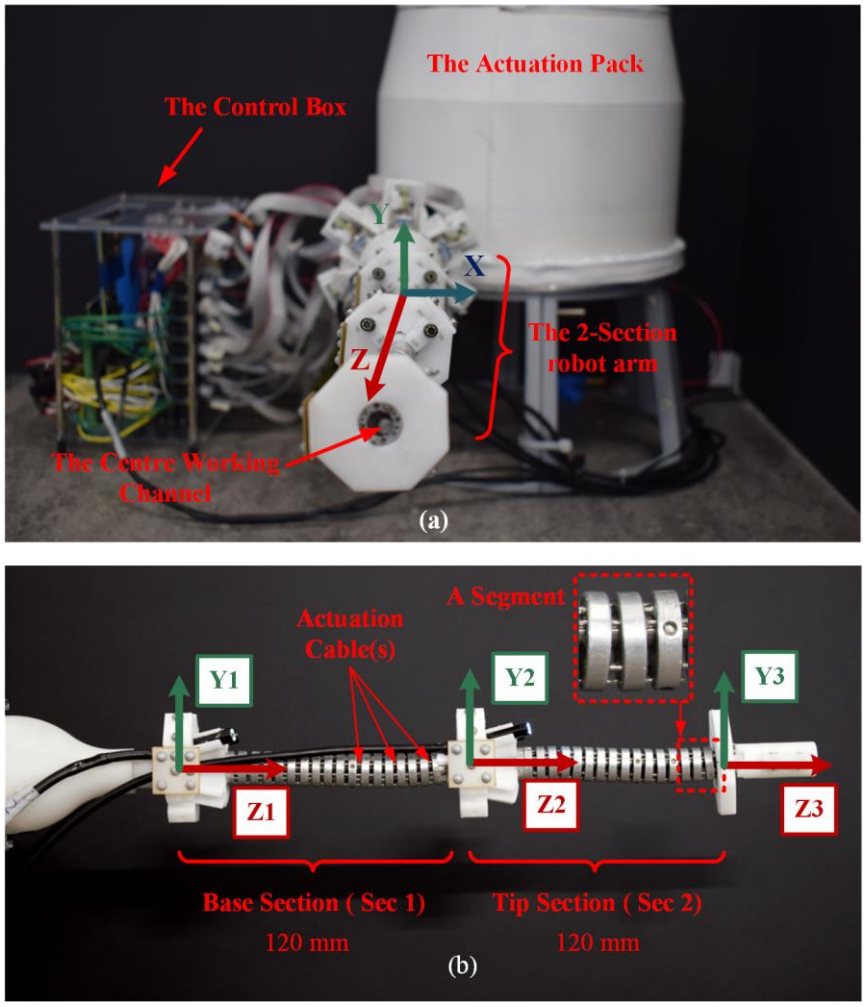


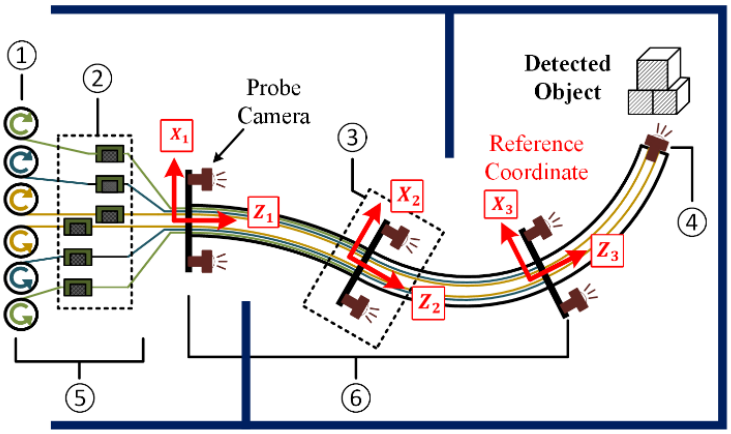
Figure 4-2 The prototype of the robot equipped with the sensing system. (a) The robot architecture. (b) The reference coordinates are defined in each section.

Figure 4-2 demonstrates the prototype of the robot equipped with the sensing system proposed above, where figure (a) shows the basic robot architecture, and figure (b)

defines the reference coordinates of the robot in each section. All the following research and experiments will be based on this scheme.

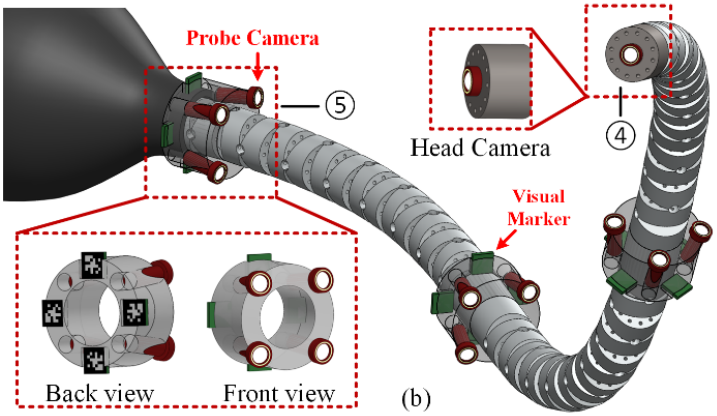
4.1 Concept of the hybrid sensing system

This section explains a new design concept of a hybrid sensing system, which consists of a dual displacement sensor system that evaluates the errors stemming from the spool cables and a self-carried visual tracking system that tracks the section tips in reference coordinates (i.e., the base of each section, as shown in Figure 4-2 (b) & Figure 4-3 (a)).



①:Actuation spools & motors (equipped with rotary encoder)
②: Linear encoder system
③: Paired observe-reference camera & marker array
④: Head camera
⑤: Dual displacement sensor system (* Formed by ① and ②)
⑥: Self-carried visual tracking system (* Formed by several ③)

(a)



(b)

Figure 4-3 The concept of a hybrid sensing system. (a) The sensing system composed of a dual displacement sensor system and a self-carried visual tracking system. (b) A 3-D sketch of the self-carried visual tracking system.

As illustrated in Figure 4-3 (a), the dual displacement sensor system (⑤) consists of rotary (①) and linear (②) encoders, where the linear encoders measure the real-time cable displacement at the proximal end of the arm, and the rotary encoders indicate the theoretical cable displacement at the motor end. By comparing these two groups of feedback values, the dual displacement sensor system works to evaluate the displacement error in the actuation pack and estimate the cable elongation in the arm structure according to the ratio of $\frac{\text{displacement error}}{\text{cable length}}$. Implemented with a novel two-stage cable displacement control approach, this device can significantly improve the control accuracy in the joint space (more details will be introduced later).

The self-carried visual tracking system (⑥) equips the continuum robot with several paired observe-reference camera & marker arrays (③) grouped by miniature probe cameras and visual markers. The probe cameras are fixed on a ring-shaped structure, tracking the visual marker attached at the backside of a front ring. These arrays are attached to the base of each section, aiming to detect the position of the section tip relative to the section base (the reference coordinate). A head camera (④) is also attached to the distal tip to observe the external environmental features and accomplish tasks such as obstacle avoidance. This tracking system helps locate the robot tips section-by-section, provides position feedback to the controller, and achieves a closed-loop control in task space.

Integrated with a novel two-level closed-loop controller (Chapter 4.2), the whole system was validated that it can successfully compensate for the control error in both joint space (cable-displacement) and task-space (tip-positioning) simultaneously, and significantly improve the control accuracy of a two-section continuum robot (Chapter Chapter 5).

4.2 The two-level closed-loop controller based on the hybrid sensing system

Precise positional control of continuum robots requires comprehensive feedback regarding the cable displacement and tip position of each section. This chapter develops a two-level closed-loop control algorithm (Figure 4-4), which consists of low-level joint-space control of actuation cable displacements and high-level work-space control of tip positions working in sequence.

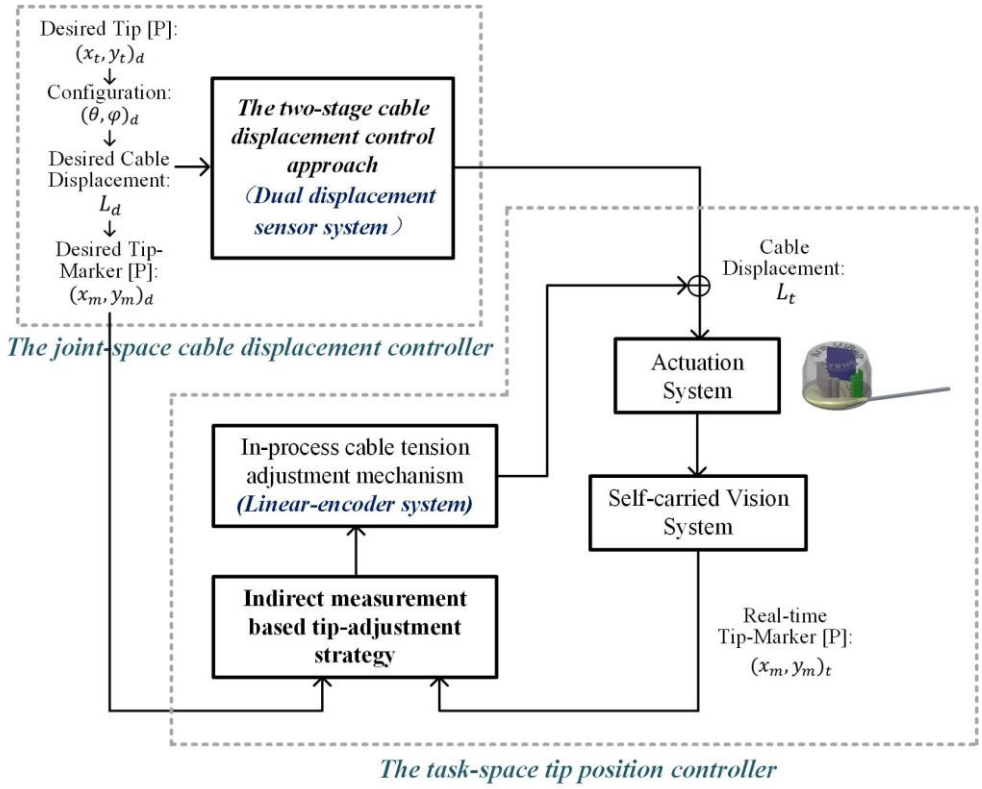


Figure 4-4 Control diagram for the two-level closed-loop controller supported by the hybrid sensing system.

The joint-space controller is based on a two-stage cable displacement control approach and the feedback from the dual displacement sensor system, aiming to minimize the displacement control error of actuation cables. Correspondingly, the work-space controller relies on the feedback from the self-carried visual tracking system and a novel indirect measurement based tip-adjustment strategy, to accurately position the tip in 3D space. An auxiliary in-process cable tension compensation approach is also

implemented. Within this framework, the robot can drive its distal tip to the desired position according to the section-by-section tip tracking, and the in-process tension adjustment approach also keeps the actuation cables appropriately tensioned and efficiently converge the tip errors.

4.2.1 The Joint-space cable displacement controller

The main error within the joint space (i.e., the displacement control of the actuation cables) is from the inexact prediction of linear motion raised by unideal cable winding and cable elongation (Figure 3-10). This section discusses the implementation of the proposed dual displacement sensor system and a novel two-stage control approach to compensate for these control errors.

4.2.1.1 The dual displacement sensor system

The dual displacement sensor system is composed of the rotary encoders placed at the motor end and the linear encoders embedded between the actuation pack and the arm of a robot (Figure 4-3 (a)). As shown in Figure 4-5, each actuation cable passes through a measurement unit before winding on the motor-spool rotary actuation. The measurement unit comprises a linear encoder chip (RLC2IC read head), a magnetic reference track, and a linear slide (THK ER513). These components are attached to a 3-D printed frame with 2 screws to clamp the actuation cables to read the displacement value in real-time. As mentioned in Chapter 3 , for the prototype robot, each single section is driven by 3 actuation cables. Thus, for the whole encoder set, 6 measurement units are mounted on a 3-D printed structure assembled with a front & rear adapter to implant with the robot system.

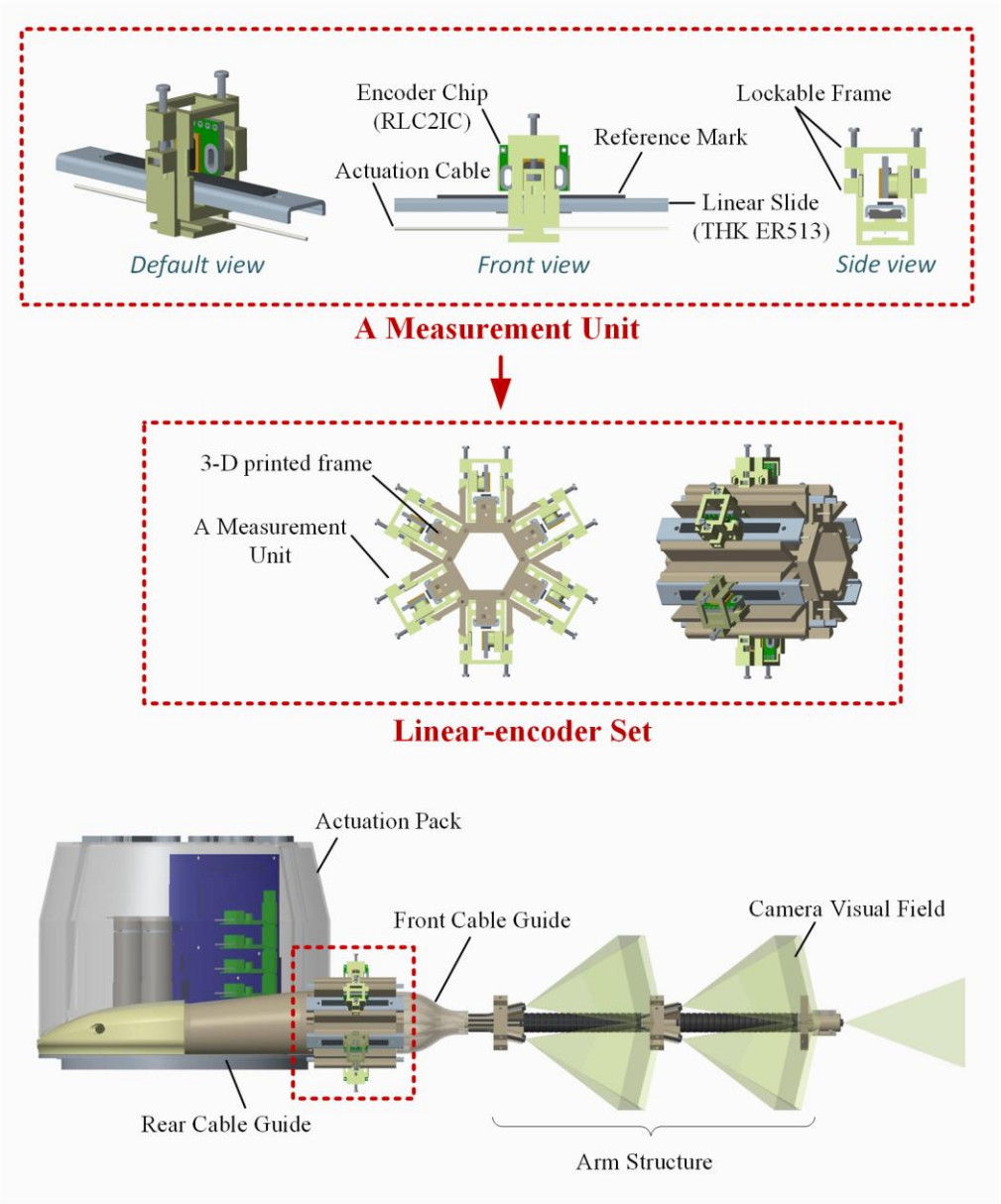


Figure 4-5 Integration of the linear encoder system: the measurement unit, the encoder set, and the whole robot system.

In practice, the rotary encoders work to indicate the theoretical cable displacement according to motor rotation angles, and the linear encoder is supposed to read the real-time cable displacement at the proximal end of the arm when the actuation cables are clamped with the encoder chips. By comparing the reading from the two types of displacement sensors, this system can precisely measure the cable displacement at the proximal end of the arm and predict the elongation in the arm structure according to known error, which essentially helps to minimize the displacement error of the whole system.

4.2.1.2 The two-stage cable-displacement control approach

In this part, a cable displacement control approach based on the dual displacement sensor system (linear & rotary encoders) is proposed to overcome the drawback of the rotary actuation design. The core idea of this approach is to divide the actuation cables into active cables and passive cables (i.e., The active cable is on the bending side and the passive cable is on the opposite side, as shown in Figure 4-6 (a)) and use different control strategies to drive these cables, followed by displacement error compensation according to the two types of displacement feedback, as shown in Figure 4-7.

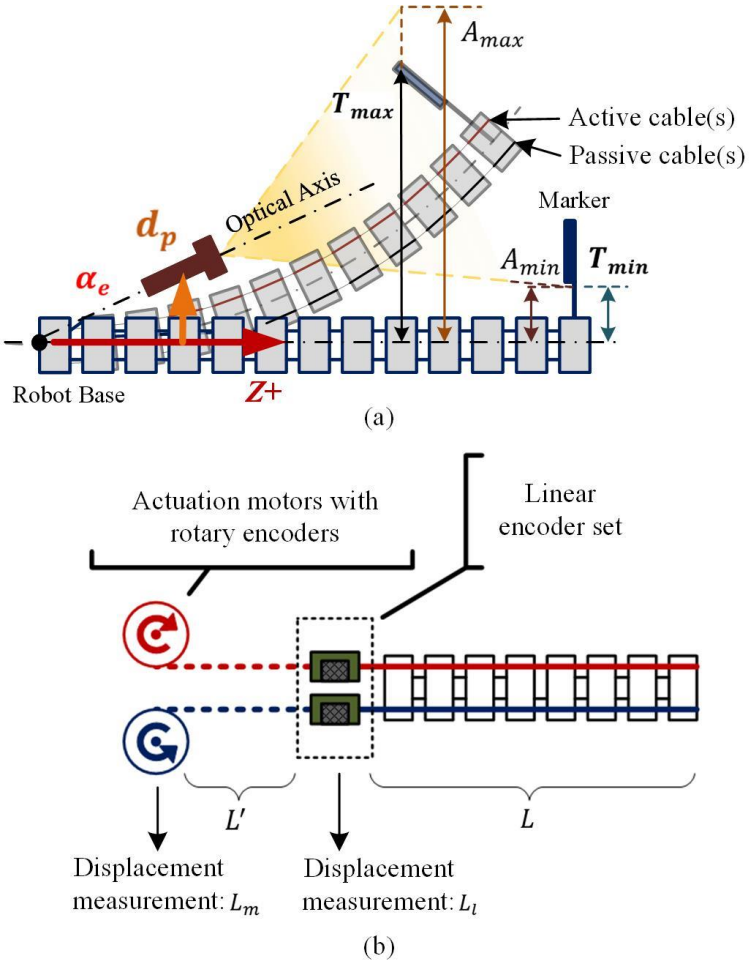


Figure 4-6 Basic definitions in joint-space control. (a) Illustration of a single-section robot with a self-carried camera. (b) The dual displacement sensor system (L_m: measurements recorded by rotary encoders; L_l: the measurements recorded by linear encoders).

Figure 4-7 shows the working principle of the two-stage cable-displacement control approach. For a specific robot bending angle θ and phase angle φ, the first-stage control

drives the cable to an approximate position. The secondary regulation predicts elongation in the arm structure and compensates for the displacement error measured in the actuation pack.

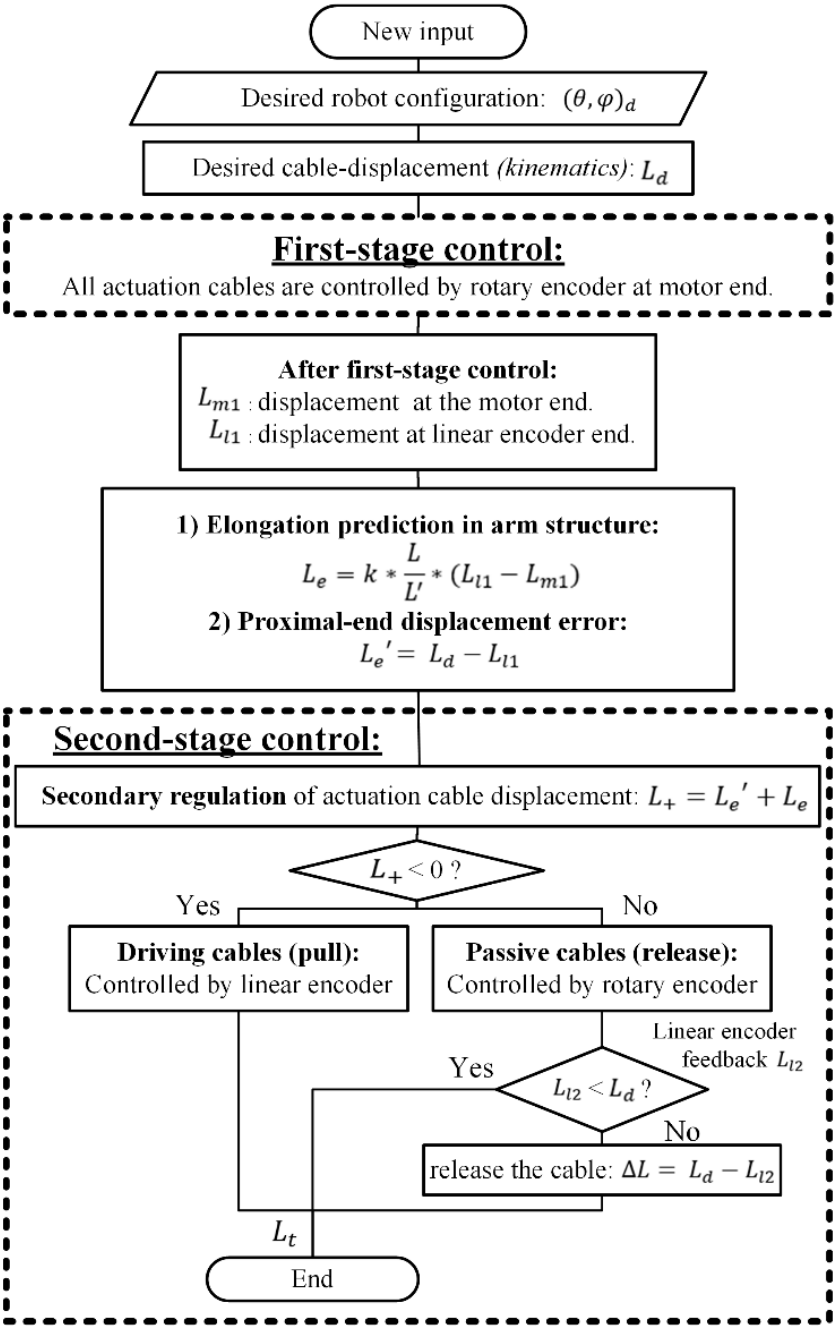


Figure 4-7 The two-stage cable-displacement control approach based on linear and rotary encoders measurement.

Detailly, as introduced in Chapter 3 , for a PCC-based continuum robot system, there are two primary control parameters for a constant curvature section, bending angle θ

and direction angle φ, the former represents the degree of the arc, and the latter indicates the rotation of the bending plane related to the axis in the arm length direction (reference). When a new command of the desired robot configuration $(\theta, \varphi)_d$ is sent to the robot controller, the theoretical (desired) cable displacement L_d is outputted based on the kinematics model.

In the first-stage control, the value of L_d is converted to motor rotation angle according to the spool parameters (Equation. (1)) to drive all the actuation cables monitored by the rotary encoders. Since then, the theoretical cable displacement at the motor end L_{m1} and the actual cable displacement at the proximal end L_{l1} can be compared, to investigate the displacement error ($L_{l1}-L_{m1}$) in the actuation pack. As mentioned before, this error comes from the unideal cable winding and elongation. Thus, the elongation error L_e in the arm can be predicted by multiplying the length ratio $\frac{L}{L'}$ with an introduced scale factor k (where L and L' are the nominal cable length in the arm structure and the actuation pack, respectively, when the arm is straight, as sketched in Figure 4-6 (b)). The value of k is empirical, and ranges within (0, 0.5).

The secondary regulation combines the displacement error measured in the actuation pack and the predictive elongation in the arm structure as a new adjustment target L_+. In this stage, the actuation cables are classified into two groups, active and passive cables. The cable lengths are denoted as L_+. The active cables have a negative relative displacement (pull cables out of the arm by spools, $L_+ < 0$), while the passive cables have positive relative displacement, as shown in Figure 4-6 a. Only the motors connecting with active cables get feedback from the linear encoder for the displacement control in the actuation space. The passive cables are controlled by the rotary encoder at the motor end, compensated by $\Delta L = L_d - L_{l2}$ (where L_d is the desired cable displacement and L_{l2} is linear encoder feedback after the second regulation for the passive cables). Finally, the actuation cables reach a displacement of L_t, which is

considered as the real-time displacement (i.e., for the driving cables, $L_t = L_d$, for the passive cables, the L_t maybe slightly differ from L_d).

This approach ensures each actuation cable keeps a proper tension and improves control accuracy & efficiency in the joint space, as validated in Chapter Chapter 5 .

4.2.2 The work-space tip position controller based on the paired observe-reference camera & marker arrays

The work-space tip position controller works to accurately position the distal tip of a continuum robot in 3D space. Basically, this controller converges the tip errors to control actions by a novel indirect measurement based tip-adjustment strategy, where the tip errors are fed back by the self-carried visual tracking system.

4.2.2.1 Design and integration of the self-carried visual tracking system

The self-carried visual tracking system is composed of several camera & marker arrays as shown in Figure 4-3). It is designed for section-by-section tip tracking to improve the control accuracy of the arm in the task-space. This part proposes a detailed system specification, including hardware design for the camera & marker arrays and software development for the visual processer.

A paired observe-reference camera & marker array is a basic unit of the self-carried visual tracking system, which consists of several forward-looking probe cameras embedded in a 3-d printed ring-shaped structure. Each single-section arm is equipped with two rings; the front ring fixed on the tip carries several visual markers, and the proximal ring set at the base works to track the corresponding tip-attached marker(s) in the reference coordinates (Figure 4-3 (b)). There are two critical parameters for the camera & marker array design, optimal camera elevation angle α_e and optimal spacing d_p (Figure 4-6 (a)). The following criteria need to be considered to keep the visual marker in sight of the camera arrays: 1) elevation angle of each camera (α_e) should be as large as possible to extend the total detection area of the camera array (especially

when the size of cameras limits the camera-view-angle); 2) an intersection area of FOVs (i.e., the field of view) between adjacent cameras needs to be settled to make sure the objective marker can transit smoothly between different camera views; 3) using a minimal number of cameras to allow a compact design for the camera & marker arrays. Thus, a simple spatial geometric model can be established to describe a single camera-marker array, where the work-space of the robot tip is presented as a circular cone and the visual fields of cameras (FOV) are presented as rectangular pyramids with fixed vertexes (i.e., a 3-D overview is shown in Figure 4-8 with a side view in Figure 4-6 (a)). The elevation angle α_e and optimal spacing d_p can be found by iteratively adjusting the angle between the rectangular pyramid and circular cone until the criteria are satisfied. Starting from the base of an arm unit, both the FOVs of the probe camera and ROM (range of motion) of the tip-attached marker can be expressed as formulas for Z+ (along the robot arm axis). Assume the single-section arm is visible for the camera ring with a maximum bending angle of θ_{max}, the ROM of tip-attached-marker can be expressed as $T \in \{T_{min}, T_{max}\}$, and the camera FOV is in range of $A \in \{A_{min}, A_{max}\}$, which can be devised as following equations:

- T_{min} and A_{min} are calculated when the robot arm is straight, $Z_+ = L_s$:

$$T_{min} = f_m - \frac{S_m}{2} \tag{23}$$

$$A_{min} = (f_c + l_c \times sin\,\alpha_e) - (L_s - l_c \times cos\,\alpha_e) \times tan\left(\frac{\alpha_v}{2} - \alpha_e\right) \tag{24}$$

- T_{max} and A_{max} are calculated when the arm reaches the maximum detectable bending angle θ_{max}:

$$Z_+ = \left(\frac{L_s}{\theta_{max}} - f_m - \frac{S_m}{2}\right) \times \sin(\theta_{max}) \tag{25}$$

$$A_{max} = (Z_+ - l_c \times \cos\alpha_e) \times \tan(\frac{\alpha_v}{2} + \alpha_e) + (f_c + l_c \times \sin\alpha_e) \tag{26}$$

$$T_{max} = \frac{L_s}{\theta_{max}} - \left(\frac{L_s}{\theta_{max}} - f_m - \frac{S_m}{2}\right) \times cos(\theta_{max}) \quad (27)$$

Where: L_s is the length of a single-section arm; f_m is the offset of the marker relative to the robot tip; S_m is the side length of the marker; l_c is the length of a probe camera; α_v is the view angle of a probe camera; f_c is the offset of the probe camera relative to the robot axis.

To calculate the optimal elevation angle α_e, an initial assumption of θ_{max} is presupposed to calculate A and T, to make sure the ROM is in the range of the camera field of view, the A_{max} must be great than or equal to T_{max} and A_{min} must be smaller than or equal to T_{min}, if these criteria cannot be satisfied, input a smaller assumption of θ_{max} until the above conditions are reached to figure out the optimal elevation of the camera relative to the robot arm α_e and the maximum detectable bending angle of the robot θ_{max} by iterative calculation. Therefore, the number of cameras needed for a single ring-structure can be calculated as:

$$x_{view} = tan\left(\frac{\alpha_h}{2}\right) \times z_+ \quad (28)$$

$$num_{cam} = ceil\left(\frac{2\pi}{2 \times atan\left(\frac{x_{view}}{Z_+}\right)}\right) \quad (29)$$

For the selected camera on this prototype robot, the optimal camera elevation angle is 18° with four probe cameras needed on each array; the maximum detectable bending of a single-section arm is 66.7°. When the bending angle of the arm exceeds the maximum detectable area, the robot will run under an open-loop mode, with no measurement and compensation of tip-positioning error, until the tip(s) are visible again. In the following experiments, a bending angle limit was implemented on the robot controller to avoid overbending.

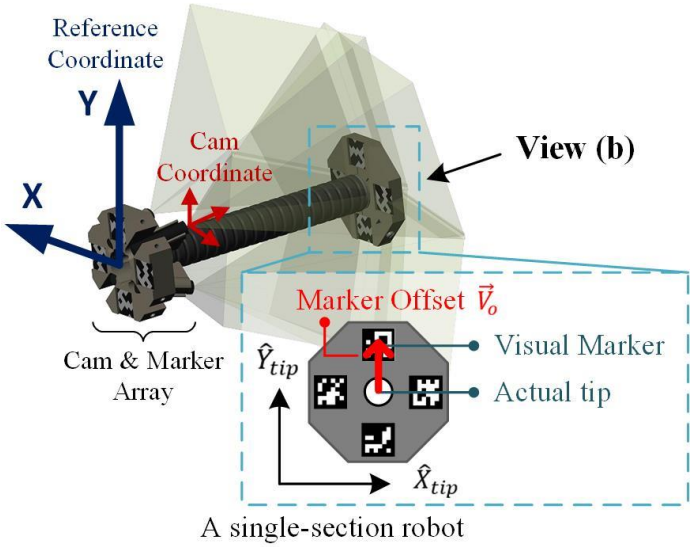


Figure 4-8 A single-section continuum robot with a paired observe-reference camera & marker array.

The visual processor processes the feedback images and outputs the position of the target markers in the reference coordinate, which is developed based on the MATLAB computer vision toolbox and OpenCV library, and the whole process includes calibration and marker tracking.

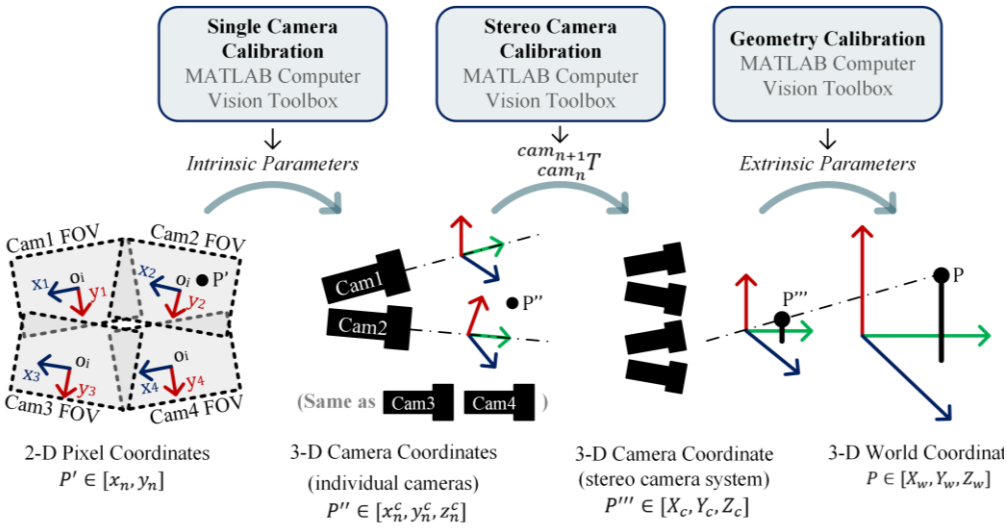


Figure 4-9 The calibration and 3-D restoring process for the visual system.

The calibration process includes 3 steps (single camera calibration, stereo camera calibration, and geometry calibration, as sketched in Figure 4-9), aiming to figure out the intrinsic & extrinsic parameters of the visual system, as well as the geometrical

relation between camera coordinate and reference coordinate [172, 173]. The single & stereo calibration processes are accomplished by MATLAB Computer Vision Toolbox directly, outputting intrinsic parameters for each individual camera and transformation matrix between each pair of adjacent cameras respectively. The extrinsic parameters between principal cameras and robot cameras are devised by the least squares method. As shown in Figure 4-10, the whole system is fixed on a straight jig, and a set of chessboard markers is sliding on the jig with a different interval. For each measurement point, the principal camera would feed back the position measurement for the chessboard T_m while the positional relation between chessboard and the robot base is known T_n, to eliminate the system error from measurements, as many as sampling points are taken, and the extrinsic parameters can be calculated by the following equation: $E_x = \frac{[T_{m1}T_{m2}\dots T_m]*[T_{n1}\,T_{n2}\dots T_n]'}{[T_{n1}\,T_{n2}\dots T_n]*[T_{n1}\,T_{n2}\dots T_n]'}$. These parameters are used to convert the 2D measurement from cameras (2-D pixel images of detected markers) to 3D position in the real world (Figure 4-9), which will be implemented in the following marker tracking tasks.

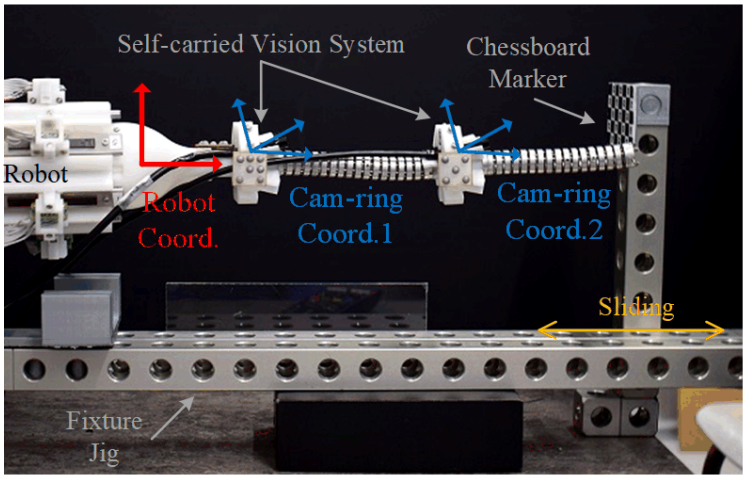


Figure 4-10 Setup for geometry calibration between robot coordinate and visual system.

Further, the section-by-section tip-tracking is based on the OpenCV library of ArUco marker detection, which is a single-marker-based visual tracking method enabling a compact design of paired camera-marker arrays (i.e., basically, ArUco markers are

designed with typical patterns, and the features of these patterns are stored in a library. Thus, when a marker is detected, its position and orientation can be restored by the pre-known features as well as the camera parameters). Under this work frame, each camera is assigned to a specific marker array with unique patterns and ID numbers, to obviate the interferences from mismatching (between markers & cameras) and other obstacles. To avoid the effects of camera blur and vibration, a camera-reading filter is also implemented in the tracking processor, which deletes abnormal readings and takes an average for valid results, making the measurement more feasible.

4.2.2.2 The indirect measurement based tip-adjustment strategy

The indirect measurement based tip-adjustment strategy is a method that converts marker error vectors to cable-displacement adjustment value, aiming to correct tip position gradually.

As mentioned previously, this method doesn't track the exact position of the robot tip but employs the positional measurement of a tip-attached marker as feedback (Figure 4-8). Detailly, as shown in Figure 4-4, when the desired tip position $(x_t, y_t)_d$ is defined in the reference coordinate of each section, the desired robot configuration $(\theta, \varphi)_d$ (arc parameters), the desired actuation cable displacement L_d, and the desired position of tip-attached marker $(x_m, y_m)_d$ are calculated according to the kinematics. The actuation cables are firstly driven by the joint-space controller (Chapter 4.2.1) to achieve displacement of L_t. Once the robot reacts and the real-time position feedback of the tip-attached marker $(x_m, y_m)_t$ is fed back by the camera arrays, the work-space tip position controller is triggered. The Cartesian positional error vector of the maker $\overrightarrow{E_m} \in R^2$ (i.e., the error between desired & actual marker position, as shown in Figure 4-11) is treated

as tip error vector[1] and used to calculate the cable-displacement adjustment value $\Delta L'_t$, gradually drive the distal tip to the desired point $(x_t, y_t)_d$.

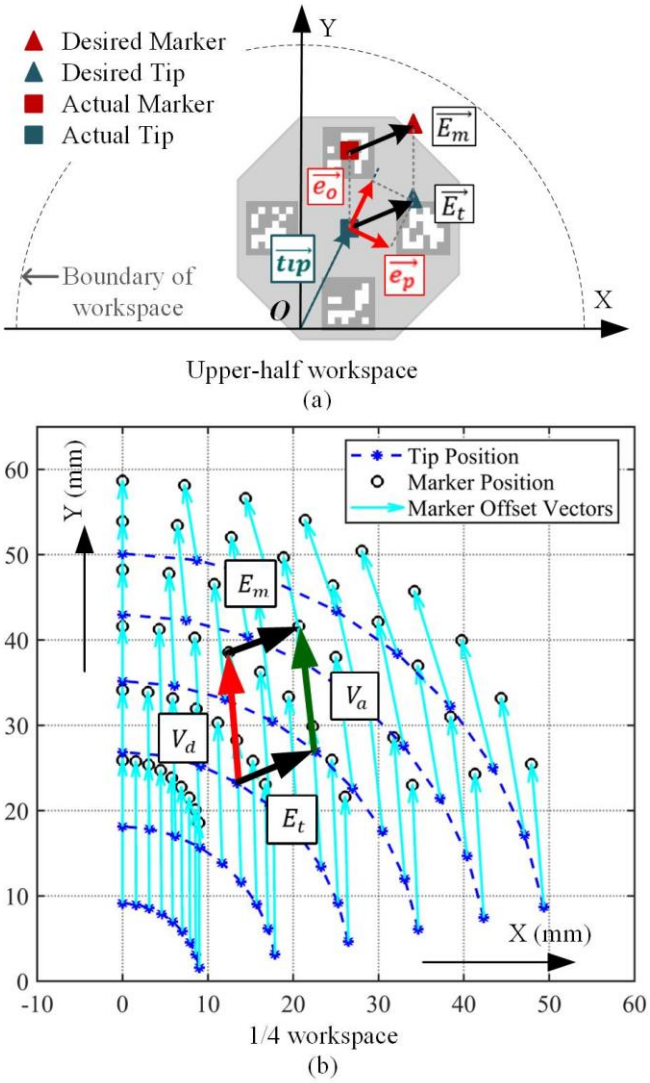


Figure 4-11 Explanation for the indirect measurement based tip-adjustment strategy. (a) A front view of the robot work-space (projection view). (b) Marker offset vectors in ¼ work-space (the blue dots: robot tip position; the black circles: visual marker position).

To substitute the exact tip error by measured marker error, this tip-adjustment strategy is developed based on the following two assumptions: 1) The actual tip-position is close

Nomenclature

[1] Theoretically, since the visual markers are fixed on robot tips with known relative position, the exact tip position could be calculated from measured marker position and orientation. However, limited by the marker size, the orientation measurement of the visual marker was not employed since its deviation is non-negligible near the singularity point (when the bending angle is small). Therefore, only the positional vector $\in R^3$ of the visual marker was received by the robot controller to calculate positioning error of markers and substitute for tip error vectors, which can also guarantee the feedback rate.

to the desired tip-position; 2) The positional error vector at the actual section tip ($\overrightarrow{E_t}$) are equivalent to the positional error vector at the tip-attached marker ($\overrightarrow{E_m}$).

According to the forward kinematics, the orientation of the section tip R_{tip} refer to the reference coordinate of each section can be expressed as:

$$R_{tip} = \left[\hat{X}_{tip} \;\; \hat{Y}_{tip} \;\; \hat{Z}_{tip}\right] = \; Rot_z(\varphi) * Rot_y(\theta) * Rot_z(-\varphi) \tag{30}$$

Where $Rot_K(\emptyset)$ denotes to rotation operator which performs a rotation about the axis $\hat{K}$ of a reference coordinate by $\emptyset$ degrees; φ and θ represent the direction and bending angle of the section.

Calculated from equation (30), the unit vectors giving the principal directions of the tip coordinate $\hat{X}_{tip}$ and $\hat{Y}_{tip}$ (Figure 4-8) can be formulated as:

$$\hat{X}_{tip} = \begin{bmatrix} (\sin\varphi)^2 + (cos\varphi)^2 * cos\theta \\ cos\varphi * cis\theta * sin\varphi - cos\varphi * sin\varphi \\ -cos\varphi * sin\theta \end{bmatrix} \tag{31}$$

$$\hat{Y}_{tip} = \begin{bmatrix} cos\varphi * cos\theta * sin\varphi - cos\varphi * sin\varphi \\ (cos\varphi)^2 + cos\theta * sin\varphi^2 \\ -sin\varphi * sin\theta \end{bmatrix} \tag{32}$$

Thus, the vector denoting marker offset $\vec{V}_o$ (Figure 4-8 & Figure 4-11 (b)) has the form :

$$\vec{V}_o = A \cdot \hat{X}_{tip} + B \cdot \hat{Y}_{tip} = [V_{o(x)} \; V_{o(y)} \; V_{o(z)}]^T \tag{33}$$

Where A and B are coefficients to determine the direction of the marker offset $\vec{V}_o$ (when the marker is on the y-axis as shown in Figure 4-8, A = 0, B=1).

Projected to the X-Y plane, $\vec{V}_{o(xy)}$ can be written as:

$$\vec{V}_{o(xy)} = [V_{o(x)} \;\; V_{o(y)}]^T \tag{34}$$

- $V_{o(x)} = A\{(\sin\varphi)^2 + (cos\varphi)^2 * cos\theta\} + B\{cos\varphi * cos\theta * sin\varphi - cos\varphi * sin\varphi\}$
- $V_{o(y)} = A\{cos\varphi * cis\theta * sin\varphi - cos\varphi * sin\varphi\} + B\{(cos\varphi)^2 + cos\theta * sin\varphi^2\}$

To model the tip-approaching process, assume the desired tip position is (θ_d, φ_d), and the current tip position is (θ_a, φ_a) (Figure 4-11 (a) & (b)), where $\theta_d = \theta_a + \Delta\theta$ and $\varphi_d = \varphi_a + \Delta\varphi$. The marker offset vectors are denoted as $\overrightarrow{V_a}$ and $\overrightarrow{V_d}$, respectively (Figure 4-11 (b)):

$$\overrightarrow{V_a} = \vec{V}_{o(xy)}(\theta_a, \varphi_a) \tag{35}$$

$$\overrightarrow{V_d} = \vec{V}_{o(xy)}(\theta_a + \Delta\theta, \varphi_a + \Delta\varphi) \tag{36}$$

In the current working space of bending angle $\theta_a \epsilon \left[0, \frac{\pi}{3}\right]$, direction angle $\varphi_a \epsilon [0, 2\pi]$, when $(\Delta\theta, \Delta\varphi) \leq 3°$, $\overrightarrow{V_d}$ and $\overrightarrow{V_a}$ has a maximum angular difference of 2.17 degrees and a maximum modular difference of 0.49 mm, which allows the $\overrightarrow{V_a}$ and $\overrightarrow{V_d}$ to be treated as parallel and equal vectors, thus $\overrightarrow{E_m}$ and $\overrightarrow{E_t}$ can also be seen as equivalent vectors. As the tip approaches the target point, the difference between $\overrightarrow{E_m}$ and $\overrightarrow{E_t}$ will be even smaller (Figure 4-11 (b)). Therefore, the tip error vector of the section $\overrightarrow{E_t}$ can be replaced by the marker error vector $\overrightarrow{E_m}$, and measured by the self-carried visual tracking system (comparing the desired & feedbacked marker position) due to the similarity.

To figure out the control action of $\Delta\theta$ and $\Delta\varphi$ (i.e., the adjustment values of robot configuration), the tip error $\overrightarrow{E_t}$ is firstly decomposed in polar coordinate along the direction of the tip vector $\overrightarrow{tip}$ into orthogonal components $\overrightarrow{e_o}$ and $\overrightarrow{e_p}$ (as shown in Figure 4-11 (a), point *o* is the center of the workspace (i.e., start point of a robot tip when the arm is straight), $\overrightarrow{E_m}$ and $\overrightarrow{E_t}$ are the positioning error of the visual marker and robot tip), and convert to $\Delta\theta$ and $\Delta\varphi$ according to following rules:

$$\Delta\theta = |\overrightarrow{e_o}| * (k_1 + k_1{}') \tag{37}$$

k_1 and $k_1{}'$ are the coefficients defined as:

$$\boldsymbol{k_1} = \frac{d(|\overrightarrow{tip}|)}{d(\theta_d)} = \frac{L}{\theta_d}(1 - cos\theta_d)$$

$$\boldsymbol{k_1}' \propto [\frac{d(|\overrightarrow{e_{o_{i-n}}}|, \dots, |\overrightarrow{e_{o_{i-1}}}|, |\overrightarrow{e_{o_i}}|)}{di}]^{-1}$$

$$\Delta\varphi = |\overrightarrow{e_p}| * (k_2 + k_2{}') \tag{38}$$

k_2 and $k_2{}'$ are the coefficients defined as:

$$\boldsymbol{k_2} = \frac{|\overrightarrow{e_p}|}{|\overrightarrow{tip}|}$$

$$\boldsymbol{k_2}' \propto [\frac{d(|\overrightarrow{e_{p_{i-n}}}|, \dots, |\overrightarrow{e_{p_{i-1}}}|, |\overrightarrow{e_{p_i}}|)}{di}]^{-1}$$

Where (k_1+k_1') and (k_2+k_2') are the equivalent proportional gain for $\overrightarrow{e_o}$ and $\overrightarrow{e_p}$, respectively. k_1 and k_2 are calculated from kinematics. $k_1{}'$ and $k_2{}'$ are relative to the convergence rate of tip error, i is the iteration number.

Finally, the cable-displacement adjustment $\Delta L_t'$ can be calculated from $(\Delta\theta, \Delta\varphi)$ via kinematics and applied to correct tip-position errors.

The tip adjustment process is terminated after all the section(s) achieved the desired control tolerance in their reference coordinates. The experiment data was presented in the reference coordinate and global coordinate to evaluate the performance of the robot in Chapter Chapter 5 .

4.2.2.3 The in-process cable tension correction approach

Additionally, within the framework of the controller, the cable tension is supervised by comparing the rotary encoder reading at the motor end and the linear encoder reading at the proximal end of the robot arm. As Figure 4-4 shows, once the cable slack is detected (i.e. reading of rotary encoder L_m is larger than Linear encoder reading L_l), the controller will give an extra negative displacement adjustment ΔL_{comp} to the corresponding actuation cable(s), where $\Delta L_{comp} = \mathrm{k_c} \times (L_l - L_m)$. Thus, the cable

can keep in tension (i.e., k_c is the compensation coefficient ranges within (0, 1), which is an empirical value and can be adjusted whenever necessary during the operation).

Overall, the indirect measurement based tip-adjustment strategy expands the sensing ability of a continuum robot system as well as saves computational resources. Cooperating with the dual displacement sensor system and the self-carried visual tracking system, the two-level closed-loop controller can converge the robot tip(s) to the desired position in local work-spaces efficiently, which largely improves the controllability.

4.3 Chapter summary

This chapter presents a comprehensive sensing system design for a two-section cable-driven continuum robot with a motor-spool actuation design. The hybrid sensing solution consists of a dual displacement sensor system working in joint space to measure actuation cable, and a self-carried visual tracking system duty in tasks space to track the robot tips section-by-section. The innovation of this system is that a secondary displacement sensor is enrolled in the actuation system of a continuum robot, thus, to assess and compensate for the multi-source cable displacement control error. Besides, it is the first time that an embedded camera device is used to indicate the robot's own parameters. Further, a two-level closed-loop controller is also developed to cooperate with the hybrid sensing system to achieve high-accuracy control in both joint space and task space.

Chapter 5 Experiments and validation for the hybrid sensing based closed-loop controller

This section presents a series of experiments to validate the hybrid sensing system on a two-section continuum robot with 4 degree-of-freedom, as an example of a multiple-section system. Firstly, the dual displacement sensor system and self-carried visual tracking systems are tested separately to evaluate sensing feasibility. Then, the synergetic sensing algorithm is integrated into the proposed closed-loop controller in task space to investigate the tip-positioning performance of this robot.

5.1 The continuum robot system setup for experiments

The robot is a two-section tendon-driven continuum arm, as demonstrated in Figure 5-1 (a). Each active section contains 20 disks with a section length of 120 mm, actuated by three steel cables. The robot employs a design of rotary actuation, all the actuation cables are coiled on the spools and then connected to the actuation motors. These motors are inherently equipped with rotary encoders (resolution: 0.09°, Figure 4-3, ①) to feedback rotation angle and predict tendon-displacement in conventional kinematics-based open-loop control, which introduces several challenges for controlling as discussed in Chapter 3.4 .

Figure 5-1 (a) & (b) demonstrates the proposed hybrid sensing system implemented on the robot. Within the dual displacement sensor system, the linear encoders are mounted at the proximal end of the arm structure, grouped by some measurement units assembled on a 3-D printed structure. Each measurement unit comprises a linear encoder chip (RLC2IC read head, resolution:2 μm), a magnetic reference track, and a linear slide (THK ER513), which can clamp with actuation cables to read the displacement value and compare with readings from rotary encoders. The self-carried visual tracking system is formed by several camera & marker arrays attached along with the arm (at the

base of each section), for section-by-section tip tracking. Some 3-D printed rings work to fix the miniatured probe cameras (USB inspection cameras with 640*480 pixels resolution) with the section base (reference coordinate), which ensures the feasibility of the visual tracking.

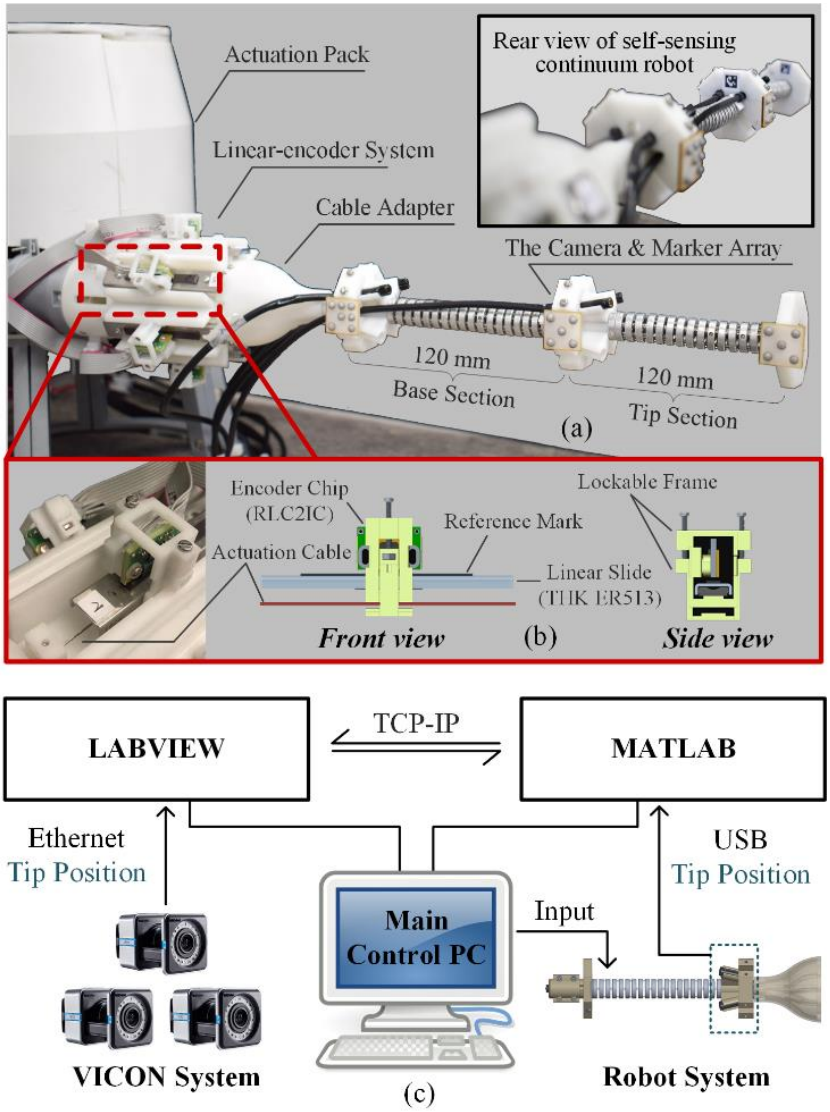


Figure 5-1 System overview. a) the two-section continuum robot with the hybrid sensing system. b) a measurement unit of the linear-encoder system, the actuation cables are clamped by the lockable frame where the encoder chip attached; c) System communication.

The whole robot is controlled by a SbRio9627 FPGA programmed in LABVIEW, and the visual processer (OpenCV) is compiled in MATLAB, communicating with the central controller via TCP-IP (Figure 5-1 (c)). Both the robot system and the visual tracking system are controlled by the main PC through LABVIEW and MATLAB

separately, and a two-way transport is established via TCP-IP between the two modules; an off-the-shelf position tracking system (VICON) runs on a separate computer and sends data to the main PC via Ethernet to provide reference data for tip-position. The maximum diameter of the robot arm is 45mm, which allows the robot to access some narrow or confined spaces with barrier-free environments to complete some tasks such as inspection and detection.

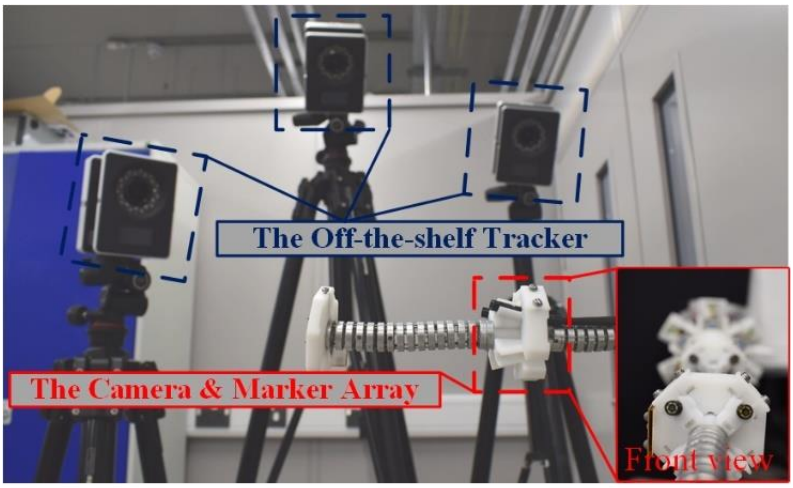


Figure 5-2 Experiment Setup: the off-the-shelf tracking (VICON) system provides a reference tip-tracking to evaluate the tip-positioning performance in the experiment.

During the experiments, an off-the-shelf position tracking system, VICON (Figure 5-2), is set aside to track the section tips in the world coordinate, aiming to evaluate the measurement accuracy of the self-carried visual tracking system and the tip-positioning performance of the robot.

5.2 Experiment results

This section presents the results of the experiment regarding the feasibility of the hybrid sensing system and the control performance of the two-level closed-loop controller.

5.2.1 Measurement accuracy test for the self-carried visual tracking system

This experiment aimed to evaluate the measurement accuracy of the proposed camera & marker array within the work-space of a single-section arm. Some sampling points

are defined along a pre-planned path in the x-y plane of the robot work-space (Figure 5-3). The robot arm is commanded to follow the path step-by-step and collect the measurements of the section tip from the off-the-shelf position tracker (VICON) and the proposed self-carried visual tracking system simultaneously. The position data feedbacked by VICON was regarded as the reference value.

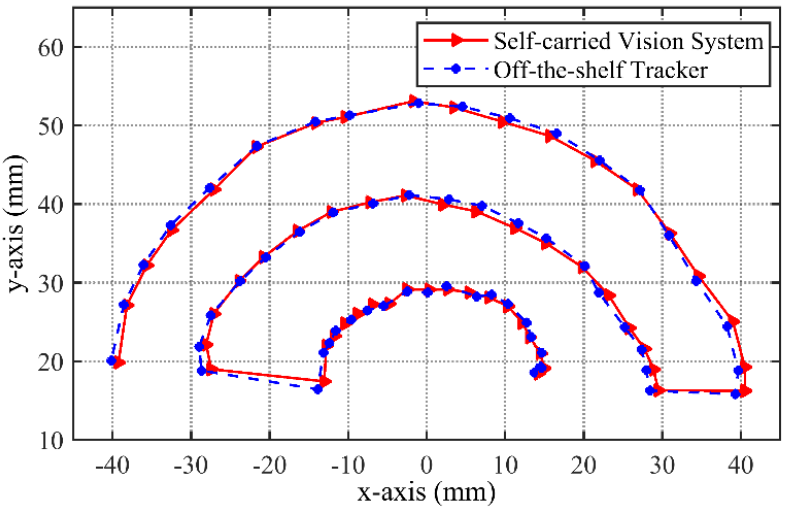


Figure 5-3 Tip-tracking performance. Tip trajectories recorded by the off-the-shelf tracker (blue) and camera & marker array (red) are compared for benchmarking the accuracy of the vision system integrated into the continuum arm.

Figure 5-3 compares the tip-recording from the VICON tracker (blue dot line) and the camera & marker array (red dot line) in the trial. The results show that the position measurement from the two sensing systems is consistent along the path, although the difference is slightly more prominent in the center area since it is the intersection area of two adjacent probe cameras.

Table 4 plots the measurement error of the vision system contrasted to the VICON Tracker. Within the 57-step trail, the mean tracking error is 0.67 mm, and the RMSE is 0.36 mm, with a feedback rate of 3 Hz. Overall, the vision system provides feasible tip-position feedback for a single-section arm with the detectable range of the camera & marker array and feedback rate to achieve closed-loop control in the actual working scenario

Table 4: Measurement accuracy of the self-carried visual tracking system compared with the Off-the-shelf Position Tracker (The Vicon System)

Min Error (mm)	Max Error	Mean Error	RMSE	Feedback rate
0.09	1.50	0.67	0.36	3Hz

5.2.2 Validation for the dual displacement sensor system based two-stage cable displacement control approach

The second experiment has two main objectives: 1) evaluating the cable-displacement control uncertainty introduced by the rotary actuation design (i.e., unideal cable winding and cable elongation); 2) examining the control performance of the new joint-space controller by comparing the tip-positioning accuracy of a 110 mm-long single-section arm under 2 types of controllers (i.e., the conventional kinematics-based open-loop controller relying on the rotary encoders and the new joint-space controller based on the dual displacement sensor system).

A butterfly path is pre-planned for the robot arm with 31 sampling points (Figure 5-5). In the first part of this experiment, the conventional model-based controller is employed, all the actuation cables were controlled by the rotary encoders at the motor end. Meanwhile, the linear encoders measured the actual cable displacement at the proximal end of the arm (Figure 5-4 (a)). Figure 5-4 (b) illustrated the cable distribution of the tested section, and Figure 5-4 (c) compares the theoretical vs actual displacements of each actuation cable in the motion. The results show a clear pattern of cable backlash (i.e., when the driving motors reverse) and elongation (i.e., the displacement error increases as desired displacement increases). Furthermore, the passive cables with positive displacement (opposite to the bending side, Figure 4-6 (a)) always have more significant control error than active cables with negative displacement (on the bending side) since the friction along with the arm structure, bringing further effect in tip-positioning accuracy.

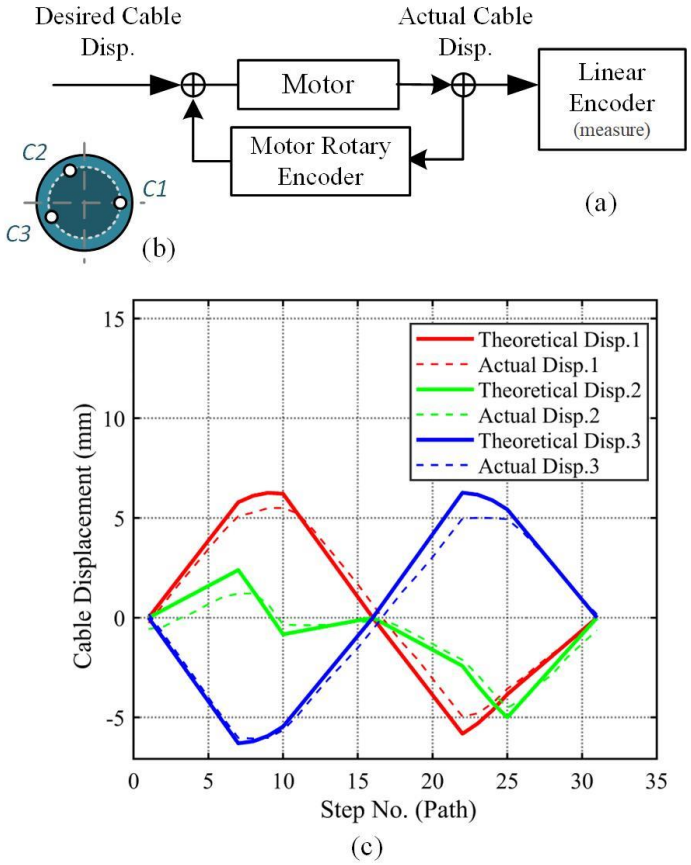


Figure 5-4 Validation for the linear encoder system. (a) workflow diagram for the experiment evaluating the rotary actuation system; (b) cable distribution for the tested section; (c) comparison between theoretical cable displacement (solid line) vs actual cable displacement (dot line).

The second part of this experiment aims to assess the joint-space cable-displacement controller (Chapter 4.2.1) by comparing it with the conventional kinematics-based open-loop controller regarding tip-positioning accuracy. The robot would repeat the same motion along the pre-planned path several times, and the self-carried visual tracking system provided tip recording simultaneously. Figure 5-5 plots the tip trajectories of the single-section arm with the conventional controller (left figure) and the new joint-space controller (right figure). The displacement error compensator at the proximal end greatly improves the performance of this robot referring to tip accuracy and repeatability. Compared with the open-loop rotary actuation mechanism, the maximum positioning error decreases from 9.96 mm to 6.70 mm, and the mean error decreases from 5.79 mm to 3.13 mm in the x-y plane.

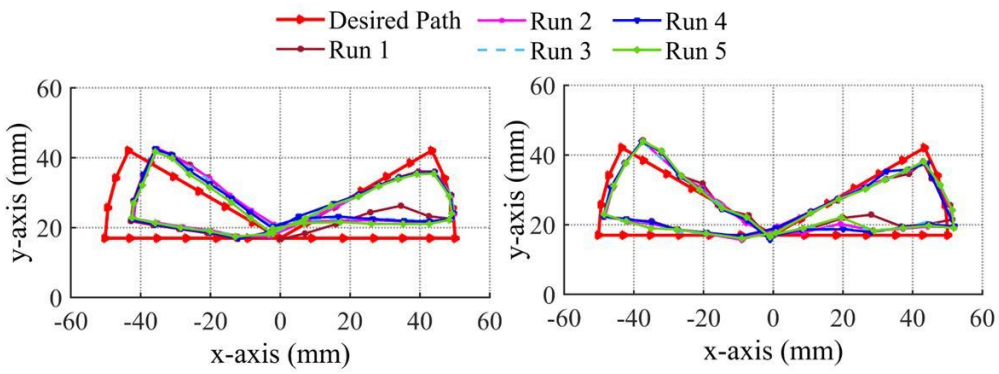


Figure 5-5 The tip-positioning accuracy comparison: open-loop control performance comparison of the single-section robot under dual-displacement sensor feedbacked control (right) and rotary encoder control (left).

Figure 5-6 compares the error vector of the above experiment, which shows that by improving the cable-displacement control accuracy, the tip error is also largely decreased, especially for corner points.

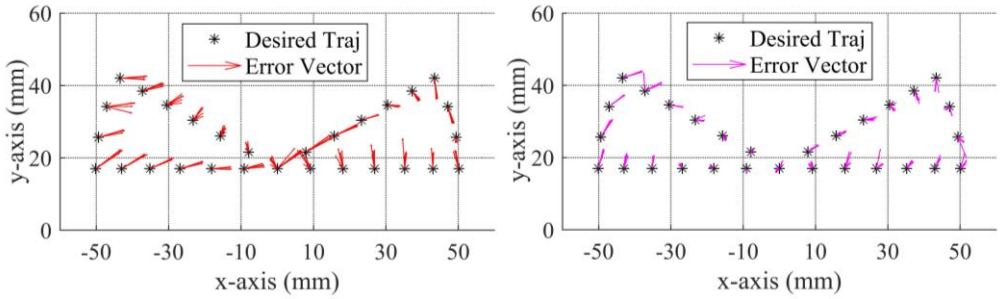


Figure 5-6 The tip-positioning accuracy comparison: error vectors comparison, under rotary encoder control (left) and dual-displacement sensor system control (right).

5.2.3 Validation for the two-level closed-loop controller with the feedback from the hybrid sensing system

The previous experiments validated the measurement accuracy of the self-carried visual tracking system and the feasibility of the joint-space cable displacement controller. This experiment examines the two-level closed-loop controller with the hybrid sensing feedback, where the robot tip is positioned by the indirect measurement based tip-adjustment strategy. As described in Chapter 4.2.2 , this controller aims to minimize the tip-positioning error of each section in its reference coordinate.

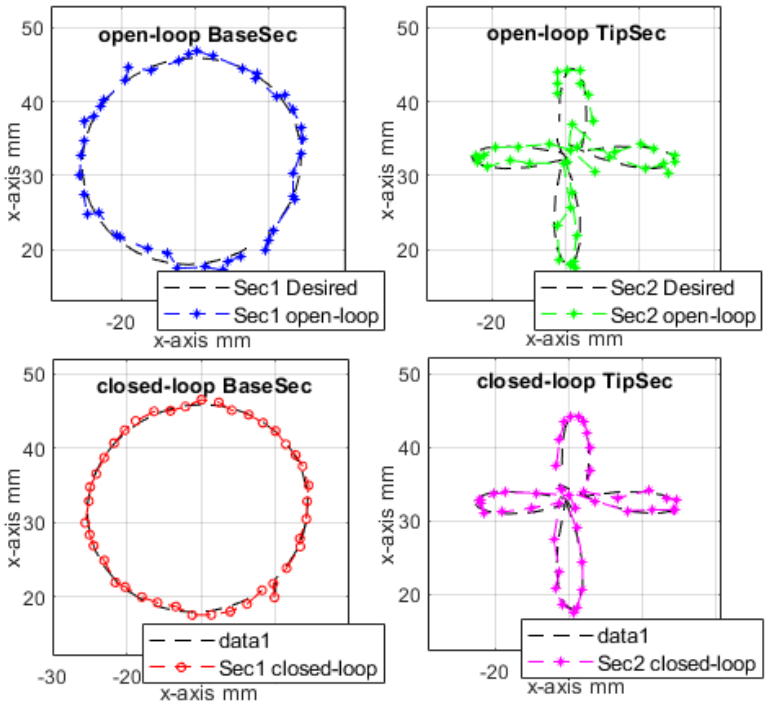


Figure 5-7 Tip-positioning performance in reference (local) coordinate: open-loop & closed-loop control accuracy comparison for both sections.

Similar to the previous tests, a continued pre-planned path is given to the robot to mimic the real working scenario like crack detection, metal welding, etc. As illustrated in Figure 5-7, the base section is expected to finish a circle path while the tip section will follow a cross-shaped path in their reference coordinates. Eventually, with the synchronous motion of the two sections, the distal tip of the robot will move in a quadrate path in the world coordinate (Figure 5-8 (a)). In this experiment, the control tolerance was set as 0.8 mm. The self-carried visual tracking system and a VICON Tracker recorded tip trajectory simultaneously to monitor control accuracy in local and world coordinates separately.

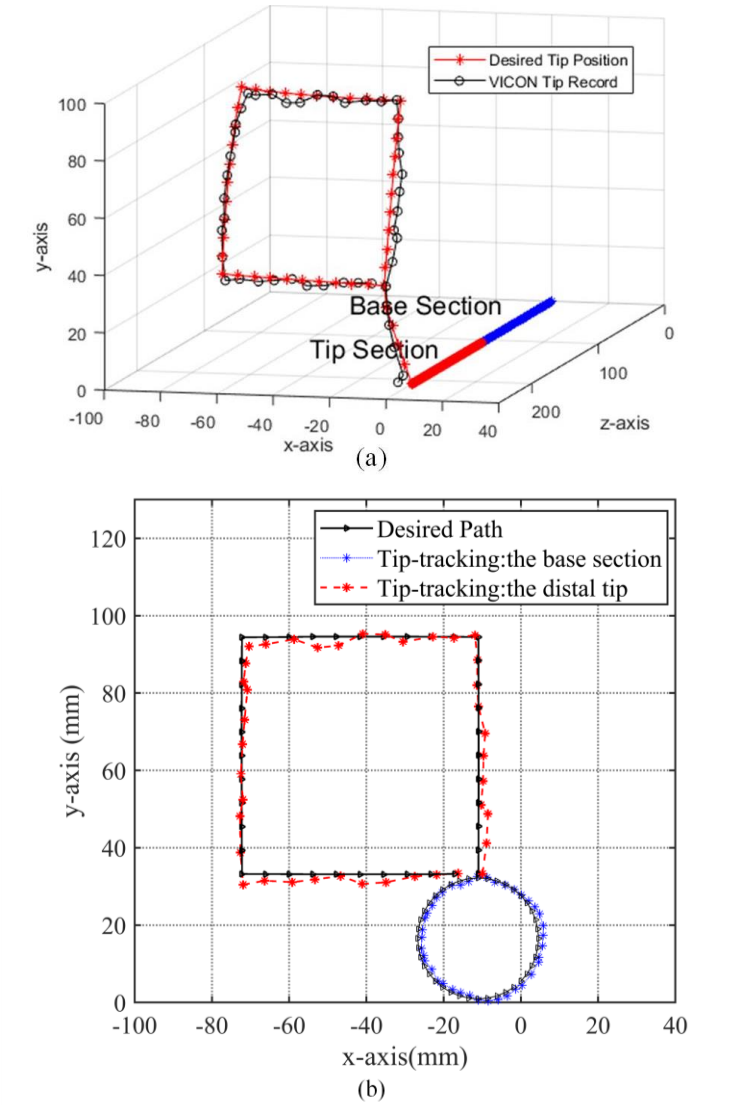


Figure 5-8 Control accuracy of the distal tip under closed-loop control supported by the hybrid sensing system in the world coordinate. (a) A 3-D view of the desired path of the desired tip. (b) A project view of the tip-trajectory in the x-y plane (black: desired path; red: the actual path of the distal tip; blue: the actual path of the base-section-tip).

As shown in Figure 5-7 and Table 5, both sections converged their tip error successfully to the desired control tolerance. The base section performs better than the tip section since its cables have no coupling problem (i.e., the cable control accuracy of a distal section was affected by the proximal section). Compared with conventional PCC-based open-loop control, the proposed closed-loop controller decreased tip-positioning error from 1.36 mm to 0.38 mm for the base section on average and reduced error by 1.64 mm for the tip section.

Table 5 Tip-positioning error for the two-section robot in the local coordinate

Error (mm)	Closed-loop error		Open-loop error	
	mean	max	mean	max
Base Sec	0.38	0.77	1.36	3.00
Tip Sec	0.59	0.80	2.23	5.77

Figure 5-8 (b) and Table 6 demonstrate the control accuracy in world coordinate monitored by the VICON tracker. Since the controller was designed for the reference coordinate operation (thus, the control tolerances were also set in reference coordinates), the global errors were more extensive than local errors. However, the error is still significantly compensated compared with the open-loop condition, e.g., the mean tip-positioning error was reduced by 1.51 mm and 4.78 mm for the base and tip sections, respectively.

Table 6 Tip-positioning error for the two-section robot in the world coordinate

Mean error (mm)	Closed-loop control error	Open-loop control error
Base Sec	0.92	2.43
Tip Sec	1.72	6.50

5.3 Chapter summary

This chapter demonstrates the experiment results and validated the hybrid sensing system as well as the two-level closed-loop controller, which improves the controllability of a multi-section continuum robot regarding preventing cable slack, minimizing open-loop control error, and increasing end-effector control accuracy in both local & global coordinates. The results show that the average tip-positioning error decreases from 6.50 mm to 1.72 mm in the world coordinate, and the local tip

positioning error in the reference coordinate for both the tip section and base section is converged to the user-input desired value (in this case within 0.8 mm). Currently, the robot achieves great tip-positioning accuracy (point-to-point) with the proposed sensing system and the new controller. The performance can be further enhanced if applying the controller to a robot system with force sensing that keeps cable tension in an ideal range.

Chapter 6 Vision-based Closed-loop Navigation for Continuum Robots in Structured Environments

The previous chapters discuss the precise tip-positioning control for continuum robots in the task space, based on the hybrid sensing system including a dual displacement sensor system and a self-carried visual tracking system. In this chapter, a led-by-the-tip navigation strategy is proposed to lead the robot to go through confined & narrow space. Cooperating with the head camera fixed on the distal tip of the robot (aiming to observe the working environment), the robot can execute some environment-interactable tasks such as obstacle avoidance and target detection.

The navigation strategy contains three subtasks: offline path planning, online obstacle detection & path optimization, and object detection. Offline path planning is the first step before the navigation truly starts, according to the desired pose of the robot body and the target region of the distal tip, a pre-planned guiding path is generated by the led-by-the-tip algorithm, then the robot system will go along with the path while the head camera is observing the working environment at the same time. If obstacles are detected, the controller will update the path instantly to make sure the robot can avoid the obstacles. Once the robot arrives at the target region, the body camera-marker arrays and the head camera will work together to measure and analyze some assigned objectives based on computer vision, which will be discussed in detail in the following sections.

6.1 Overview of the navigation strategy

In this section, the setup of the robot system prepared for the navigation task will be presented as well as a flowchart of the control strategy.

6.1.1 The robot system setup for navigation

For the navigation tasks, the 2-section prototype robot (Figure 6-1) is installed on a 200-cm linear rail (Igus, Drylin) to provide an additional linear motion in the Z direction as shown in Figure 6-1. The whole system has 5 degree-of-freedom in a 3-D workspace (i.e., each section of the robot has 2 degree-of-freedom in the X-Y plane, and the linear rail provides linear movement in the Z direction). A head camera is also added to the robot's vision system (attached to the distal tip of the robot). Unlike the self-sensing camera-marker array introduced before, the head camera provides outward sensing and works to observe the working environment. For example, during the inserting process, the head camera can locate the obstacles against the robot, providing environmental information to the robot controller to update the navigation path online.

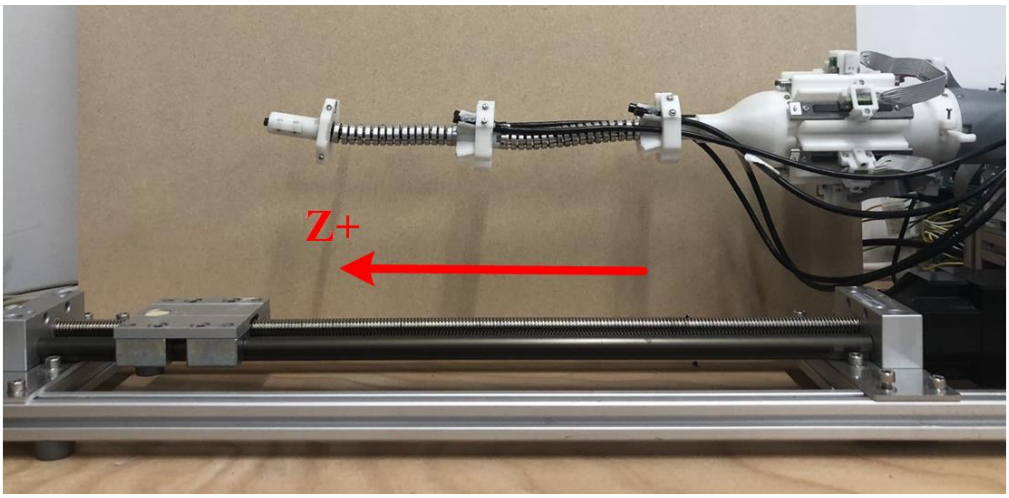
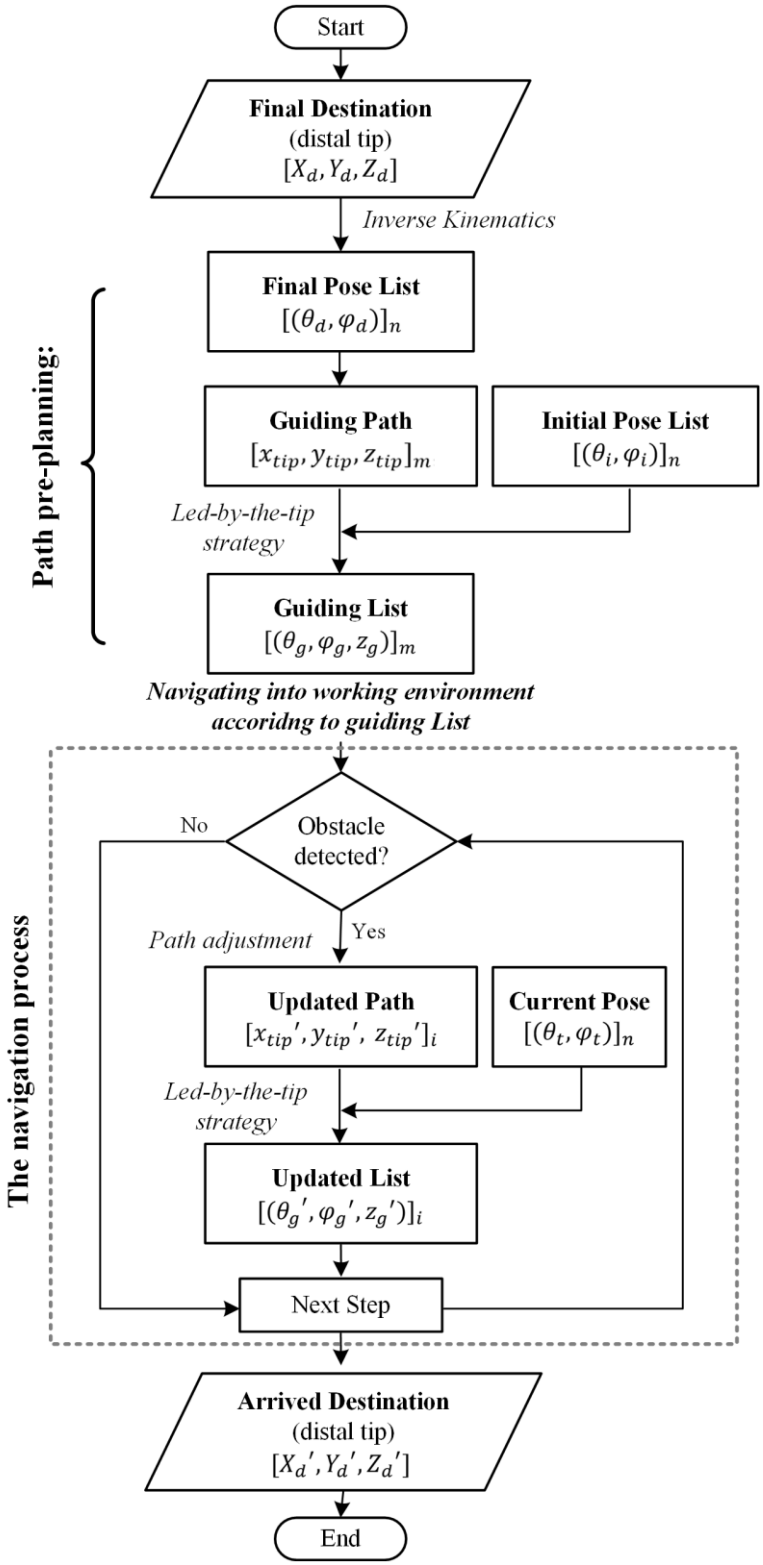


Figure 6-1 The robot system setup for the navigation task. The 2-section snake arm is mounted on a linear rail to provide linear motion.

6.1.2 The control strategy of robot navigation

As shown in Figure 6-2, the control target is to navigate the robot's distal tip to a desired destination $[X_d, Y_d, Z_d]$, refer to a user-defined world coordinate (usually at the base of the robot system).



*Footnote subscript: $\boldsymbol{n}$ – number of sections of the robot; $\boldsymbol{m}$ – step numbers of the output path.

$\boldsymbol{i}$ – step number of the updated path.

Figure 6-2 The work flowchart for the navigation process, which is divided into 2 stages: path pre-planning and navigating into the working environment. In the first stage, a guiding list is generated

according to desired tip position; for the second stage, the robot goes into the working environment while functions like obstacle avoidance are achieved.

Firstly, the target destination $[X_d, Y_d, Z_d]$ is converted to the desired robot pose $[(\theta_d, \varphi_d)]_n$ by iterative approaching inverse kinematics (i.e., in Chapter 3.2.2 , n is the number of sections of the robot). Then, in the stage of path pre-planning, an m-step initialized guiding list $[\theta_g, \varphi_g, Z_g]_m$ is exported based on the led-by-the-tip strategy (i.e., θ_g & φ_g is the optimized robot pose at each step, Z_g is the forward motion of the robot along Z-direction, provided by the linear rail). This guiding list is supposed to lead the robot to go forward and snake its way. During the navigation process, the camera continuously updates the environmental information to the robot controller, if potential obstacles are detected, the guiding path will be re-planned online according to the position of obstacles and the instant robot pose $[(\theta_t, \varphi_t)]_n$, which makes the robot can successfully pass through the working channel. Finally, the after reaching the final destination $[X_d', Y_d', Z_d']$ along the updated guiding list $[\theta_g', \varphi_g', Z_g']_i$, the robot will finish the assigned measurement and detection tasks by the self-carried vision system (i.e., the arriving destination may vary from the initial input of $[X_d, Y_d, Z_d]$ to avoid the possible obstacles). All the processes are introduced in detail in the following chapters.

6.2 Stage 1 – Path pre-planning: the led-by-the-tip strategy

As described in Part 6.1 , this stage aims to generate an initialized guiding list $[\theta_g, \varphi_g, Z_g]_m$ to navigate the robot reaching the target destination $[X_d, Y_d, Z_d]$ with final pose of $[(\theta_d, \varphi_d)]_n$.

6.2.1 Discretized guiding path

To begin with the path pre-planning, the curve representing the final pose of the robot $[(\theta_d, \varphi_d)]_n$ is discretized to an m-step guiding path $[x_{tip}, y_{tip}, z_{tip}]_m$ first (Figure 6-3 & Figure 6-4). Ideally, the distal tip will move along this path, and the robot body will follow the distal tip to approach the target destination.

In detail, each section with pose $[\theta, \varphi]$ is divided into q subsections according to feed in step length f defined in z direction (i.e., step length of the linear rail) and the section length of the robot, where:

$$q = \frac{s}{f} \tag{39}$$

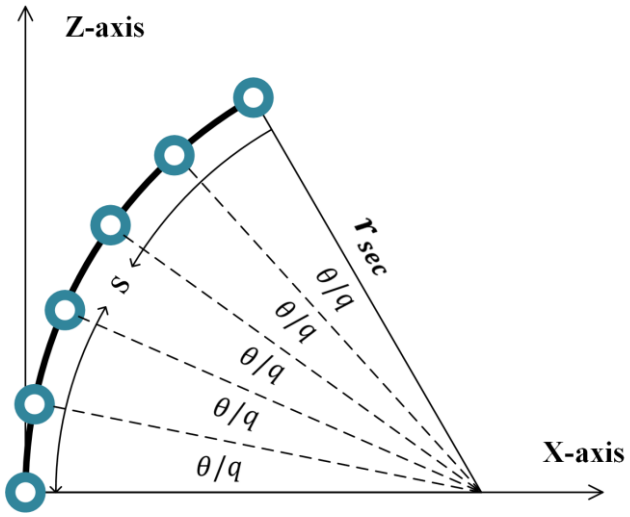


Figure 6-3 Discretization for a single-section continuum robot.

An example of a discretized single section is sketched in Figure 6-3, and the expressions of nodes on the curve can be deduced based on the kinematics model introduced in Chapter 3 :

$$r_{sec} = \frac{s}{\theta} \tag{3}$$

$$P_{(j)} = \begin{bmatrix} x_{(j)} \\ y_{(j)} \\ z_{(j)} \end{bmatrix} = \begin{bmatrix} r_{sec} \times (1 - \cos(\frac{\theta}{q} \times j)) \times \cos\varphi \\ r_{sec} \times (1 - \cos(\frac{\theta}{q} \times j)) \times \sin\varphi \\ r_{sec} \times \sin(\frac{\theta}{q} \times j) \end{bmatrix} \tag{40}$$

Where $j \in (0,1,2,\dots,q)$

Similarly, the multiple section problem can be solved by equations (7) to (9). Finally, the number of steps of the discretized guiding path can be calculated as:

$$m = n * q + 1; \tag{41}$$

Figure 6-4 shows an example of a 49-step guiding path for the 2-section prototype robot, within which, the feed-in step length is defined as 5mm.

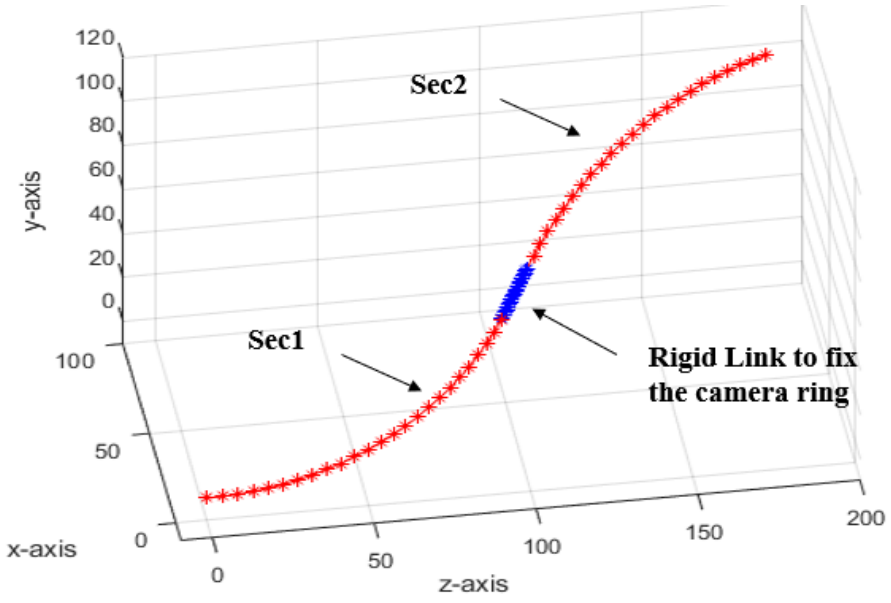


Figure 6-4 The discretized m-step guiding path $[\boldsymbol{x_{tip}}, \boldsymbol{y_{tip}}, \boldsymbol{z_{tip}}]_{\boldsymbol{m}}$ for a 2-section prototype robot.

6.2.2 The led-by-the-tip strategy

As shown in Figure 6-2, the led-by-the-tip is a control strategy that let the section tip(s) (snake body) of the robot follow the trajectory of the distal tip, while the distal tip goes forward along a guiding path $[x_{tip}, y_{tip}, z_{tip}]_m$. The aim is to output a guiding list describing the robot's pose $[\theta_g, \varphi_g, Z_g]_m$ (control angles) at each step to match the pre-planned guiding path.

The input of the algorithm is the guiding path for the distal tip $[x_{tip}, y_{tip}, z_{tip}]_m$ and the initial or starting pose of the robot $[(\theta_i, \varphi_i)]_n$. In practical work, the robot always starts at a straight state (i.e., no bending), namely, when the value of θ and φ for all the sections is zero, as illustrated in Figure 6-5. **The core idea is to establish a local optimization problem to find the optimal robot arc parameters and feed-in**

distance $[\theta, \varphi, z]_n$ step-by-step to fit the robot pose with the shape of planned path, an example of the desired robot pose at a specific step is given in Figure 6-5.

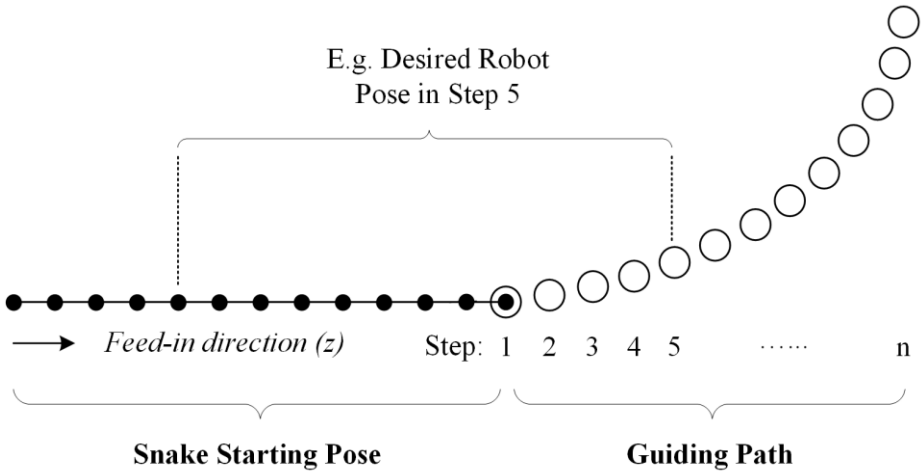


Figure 6-5 Illustration for the starting pose of the robot (black points), desired guiding path (white circle) and desired robot pose for a specific step.

The objective function in this problem compares the shape of the predicted robot pose with the pre-planned path, which is formed by two elements, the tip error E_{tip} and the body deviation E_b, weighted by coefficients w_1 and w_2 (equation (42)), and the method of local search will find the optimal solutions of $[\theta, \varphi, z]_n$ with minimum value of shape difference. Usually, w_1 has a greater value than w_2 since the path-following performance of the distal tip is a more significant criterion than the trajectory-following performance of the snake body, when evaluating the navigation.

$$Fun(\theta, \varphi, z) = w_1 \times E_{tip} + w_2 \times E_b \tag{42}$$

To simplify the shape comparison, the predicted robot pose is represented by a set of sampling points with the number of $(q * n + 1)$ (i.e., each section is discretized into q subsections, similar to the discretized guiding path, so the predicted robot has the same resolution with the guiding path, as shown in Figure 6-6). The Cartesian coordinate of these sampling points can be calculated from the guessed value of $[\theta, \varphi]_n$ by equation (40), plus the feed-in distance z at each point.

Within equation (42), the tip error is simply the Euclidean distance between the tips of the predicted and desired robot pose (Figure 6-6).

$$E_{tip} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \tag{43}$$

The body deviation is represented by the maximum Euclidean distance between all the pairs of sampling points (in the case of Figure 6-6, the body deviation is the length of the longest blue dotted line).

$$E_b = \max\left(E_{b1}, E_{b2}, E_{b3}, E_{b4}, \dots, E_{b(q*n)}\right) \tag{44}$$

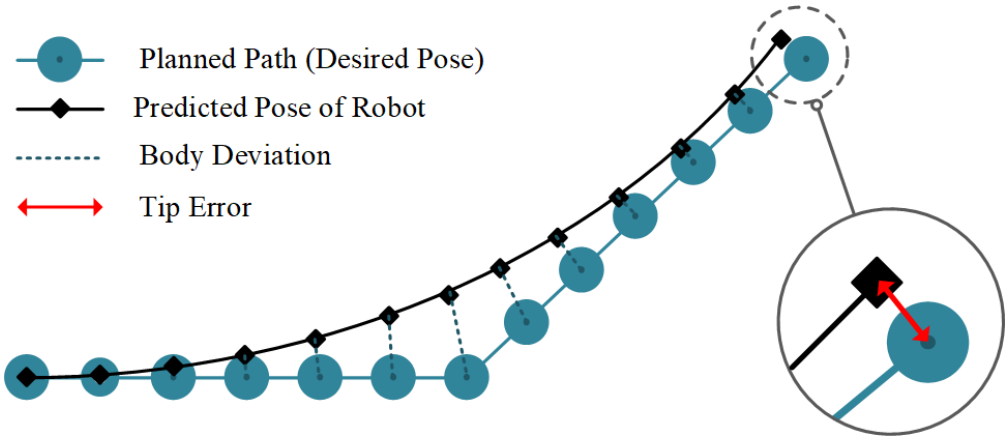


Figure 6-6 Sketch of robot pose fitting. The predicted robot is discretized into several subsections and represented by a group of sampling points with the same resolution as the planned path.

For each step, the search for the optimal solution starts from the current robot pose, the predicted value is iteratively attempted until the minimum value of the objective function is found. To minimize the computing time of optimization, the input criteria are defined as bending angle $\theta \in (0°, 90°)$, direction angle $\varphi \in (0°, 180°)$, feed-in distance $z \in (0\ mm, f)$(i.e., f is the user-defined maximum step length). The search space (constraints) is limited to a range of $\pm 6°$ for robot angles and $\pm 5\ mm$ for feed-in distance, these values are suggested by geometry calculations considering possible tip error and body deviation. Finally, an m-step guiding list indicates the robot pose for each step $[\theta_g, \varphi_g, Z_g]_m$ is output.

The process of path pre-planning is developed and tested in MATLAB with the optimization toolbox, using the method of constrained non-linear optimization. The parallel computing toolbox is also involved to accelerate the process.

Continue with the example shown in Figure 6-4, the navigation process of the 2-section prototype robot can be simulated as follows:

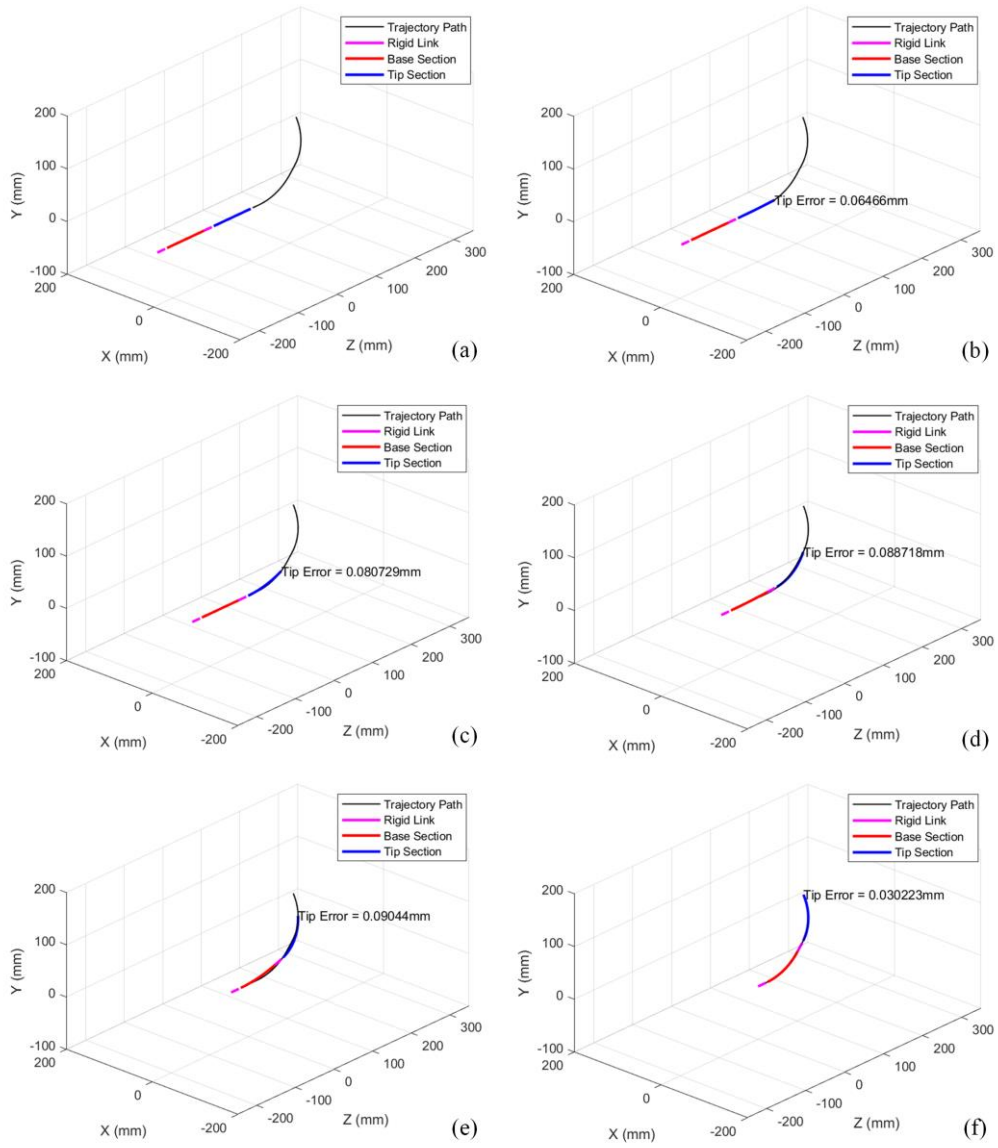


Figure 6-7 Simulation of the navigation process based on the guiding list $[\theta_g, \varphi_g, Z_g]_m$, the solid line in black represents the pre-planned path for the robot, the blue arc illustrates the tip section, and the red arc presents the base section, the pink straight lines represent the camera & marker arrays at the base of each section. (a) step 1: initial pose of the snake. (b) step 10. (c) step 20. (d) step 30. (e) step 40. (f) step 49: the final pose of the robot at the end of the navigation, fully inserted into the desired path.

6.3 Stage 2 - Vision aided navigation with on-line obstacle avoidance & path re-planning

In the second stage, the snake robot is supposed to be navigated into the working environment step-by-step according to the guiding list $[\theta_g, \varphi_g, Z_g]_m$, meantime, the self-carried camera-marker arrays work to correct the robot pose and the head camera observes the surrounding. Once the obstacle is detected and its distance to the guiding path is smaller than the safety distance, the path adjustment process (re-planning) will be triggered to shift the path according to the user-inputted adjustment direction (the direction pointer in the figure) to prevent oncoming dangers.

6.3.1 Obstacle detection

Figure 6-8 describes the principle of objective detection work (which could be derived from basic computer vision theory [174, 175]). When navigating the robot arm into a working channel, the head camera stands by to observe the surrounding features. When a potential obstacle occurs in the visual field of the camera, the location of this obstacle ${}^{Cam}_{Obs}P$ against the camera coordinate will be commented (i.e., calculated from the camera intrinsic parameters and pixel coordinate of the objective in the image plane). To check the minimum distance from the obstacle to the guiding path $[x_{tip}, y_{tip}, z_{tip}]_m$, the ${}^{Cam}_{Obs}P$ need to be converted into the world coordinate:

$$ {}^{World}_{Obs}P = {}^{World}_{Cam}T \times {}^{Cam}_{Obs}P + {}^{World}_{Cam}P \tag{45} $$

Where:

$$ {}^{World}_{Cam}T = {}^{World}_{Tip}T \times {}^{Tip}_{Cam}T = \begin{bmatrix} {}^{World}_{Cam}R & {}^{World}_{Cam}P \\ 0\ 0\ 0 & 1 \end{bmatrix} \tag{46} $$

Since the robot pose is controlled by the closed-loop controller proposed in Chapter 4 and its tip-positioning accuracy has been proved, the deviation between the actual and desired robot pose can be neglected. Thus, the ${}^{World}_{Tip}T$ can be directly calculated from the theoretical robot pose extracted from the guiding list $[(\theta_g, \varphi_g, z_g)]_m$ for a specific

step, applying the kinematics stated in equation (3) to (9) (Chapter 3 . The ${}^{Tip}_{Cam}T$ is a constant value figured out by geometry calibration between the head camera and the robot system using the least square method, which is presented in Chapter 4.2.2 .

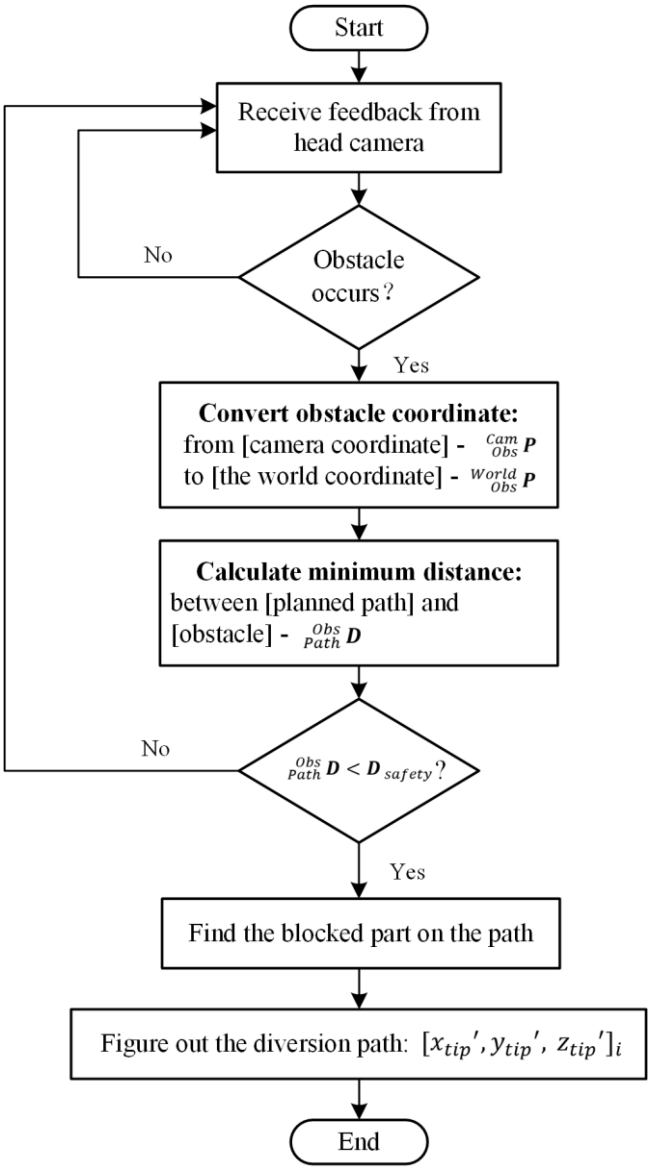


Figure 6-8 Working principle for obstacle detection.

Once the position of the detected obstacle against the world coordinate ${}^{World}_{Obs}P$ is calculated, the minimum Euclidean distance from the obstacle to the guiding path $[x_{tip}, y_{tip}, z_{tip}]_m$ can be searched and stored as ${}^{Obs}_{Path}D$. Compared with the user input safety distance D_{safety}, the value of ${}^{Obs}_{Path}D$ can indicate whether the guiding path is

blocked. If so, the process of on-line path replanning will be triggered to adjust the rest of the path to avoid the robot from the obstacle.

In the general working scenarios of continuum robots, the detection objectives are mainly the boundaries or barriers in the working environment, thus, various visual recognition algorithms can be applied and explored to better recognize this kind of characterization.

6.3.2 On-line path replanning

As illustrated in Figure 6-2 and Figure 6-9, once the potential obstacles are detected that may block the guiding path, the controller will shift the original planned path to an appointed direction (prior information, defined by user input), a new diversion path $[x_{tip}', y_{tip}', z_{tip}']_i$ is generated (i is the step number rested, i.e., if a pre-planned path has 10 steps in total and currently the robot is in step 3, then, there are 7 steps rest to adjust the path).

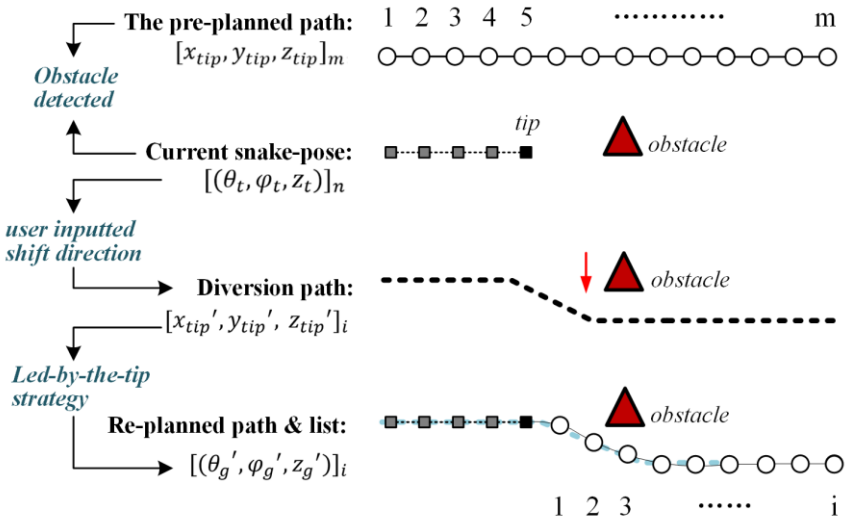


Figure 6-9 Method of path re-planning. When an obstacle occurs in the pre-planned path and is detected by the snake robot at the current location, the controller will re-direct the planned path according to user input to avoid the obstacle.

Detailly, as shown in Figure 6-9, the new guiding list $[\theta_g', \varphi_g', Z_g']_i$ is updated by the pose fitting method presented in the led-by-the-tip strategy (6.2.2), based on the shifted path $[x_{tip}', y_{tip}', z_{tip}']_i$ and current robot pose $[(\theta_t, \varphi_t, z_t)]_n$. The robot will go

forward along the new path to avoid hitting the obstacles until the next obstacle occurs. However, after path re-planning, the arriving point of the distal tip may differ from the original desired destination.

6.4 Stage 3 - Target detection and measurement

Furthermore, after arriving at the target destination $[X_d', Y_d', Z_d']$ (i.e., the arriving destination may vary from the initial input of $[X_d, Y_d, Z_d]$ to avoid the possible obstacles), the embedded visual system is assigned to finish some tasks such as detection & measurement, based on the theory of computer vision.

6.4.1 The working frame of target measurement

The work of target measurement is divided into 2 steps: online image capture and off-line image processing:

- Step1: Image capture from head camera (online)
    - A new TCP-IP channel is built between LabVIEW and MATLAB (Figure 6-10), aiming to transfer robot parameters (from LabVIEW) to the image processor (MATLAB).
    - During the navigation (when the robot is operated), capture the view of the head camera whenever the target object occurs.
    - Store the images captured in the form of structures: {image, current robot configuration, current linear position z}.

After collecting all the images needed, the image processor starts to work to analyze the pictures.

- Step 2: image processing for object measurement (offline)
    - The processor starts to work after the navigation finishing.
    - The images are compared and analyzed to measure the dimension of an object via the method of triangulation (which will be introduced later).

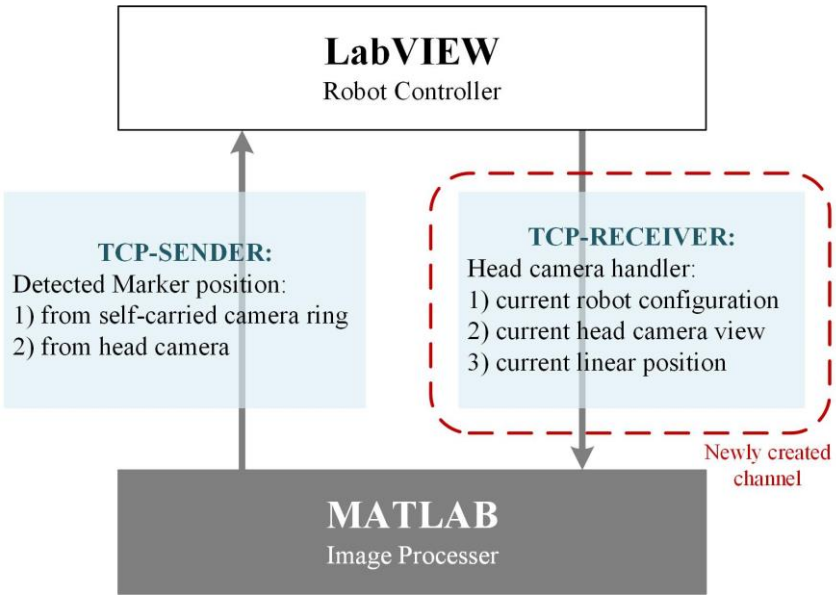


Figure 6-10 Communication channels between MATLAB and LabVIEW for the task of object measurement.

Figure 6-10 illustrates the data communication scheme between the LabVIEW robot controller and MATLAB image processor, which also lists all the data needed for the task of object measurement.

6.4.2 Method of pseudo-triangulation based on a single (head) camera

Linear triangulation is the method to restore the 3-D features from multiple 2-D images [175]. Usually, the method is applied by using multiple (at least 2) cameras with known positions to take pictures of an object from different views. In this application, the object is observed by a single camera --- the head camera on the robot.

Figure 6-11 describes the working process of the pseudo-triangulation method. To mimic the working principle of linear triangulation, the camera will capture images of the object via various perspectives, while the configuration of the robot is recorded. The position and orientation of the camera at different moments (corresponding to the captured images) are calculated by the kinematics model according to the robot configurations tracked by the self-carried camera & marker arrays. Thus, the 3-D feature points can be restored by the Vision Toolbox named ‘triangulate’ and ‘triangulateMultiview’ are used. The input is the images, the camera matrix, and the

corresponding camera positions & orientations. The output is the 3-D position $[x, y, z]$ of objective points.

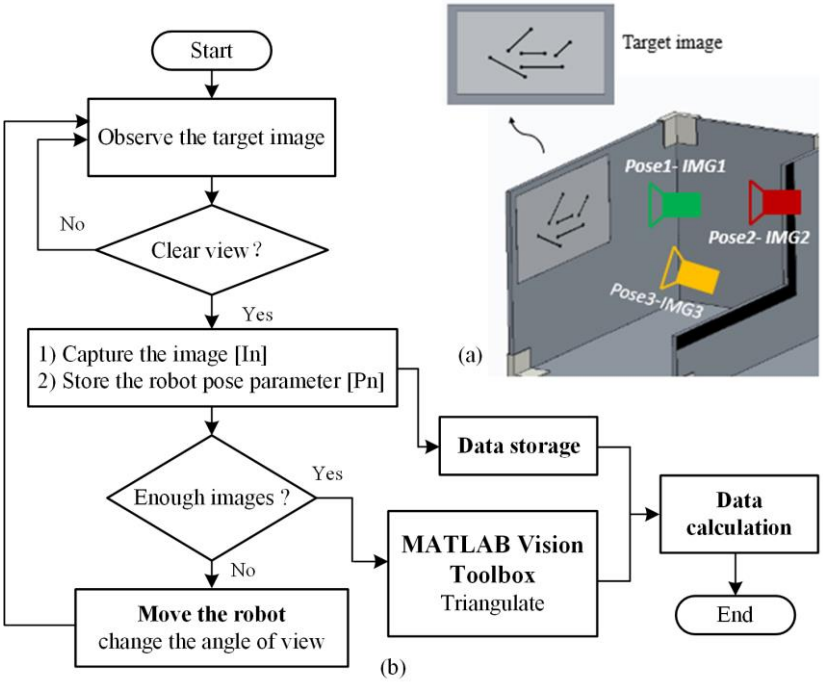


Figure 6-11 Flowchart of the pseudo-triangulation method based on the head camera. (a) An animation. (b) The working principle of the triangulation process.

6.5 The remote controller developed for navigation tasks

The continuum robot system is designed to work in extreme and hazardous environments, to ensure safe operation, a remote-control scheme is proposed. The core idea is to develop an App running on an Android Tablet, which works to send control instructions to and receive robot parameters from the main control PC remotely via TCP-IP. This controller Application is developed in Android Studio based on JAVA language (Figure 6-12).

Figure 6-13 demonstrates the user interfaces of the entry page and navigation bar of this controller, which shows the two working modes of this App: controller mode and navigation mode.

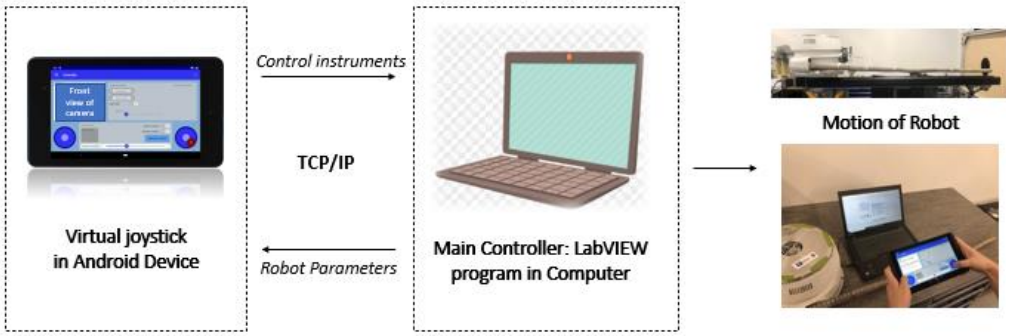


Figure 6-12 An overview of the controller App: the data communication, the working scenario, and a working demo.

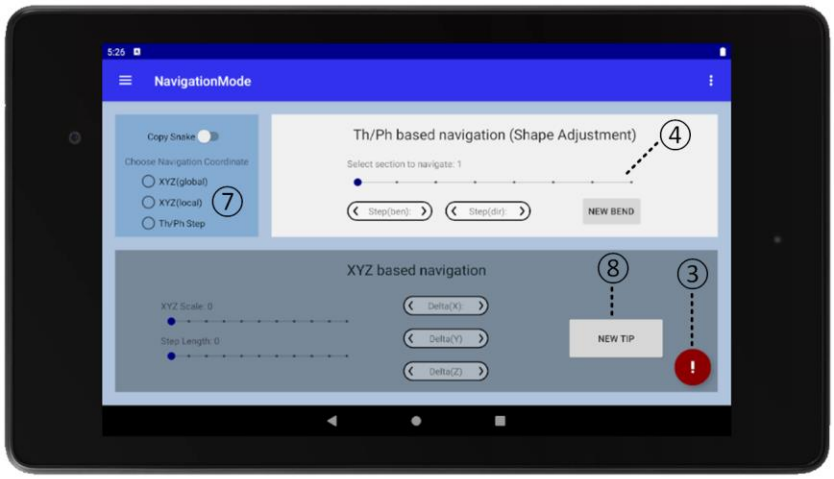


Figure 6-13 The user interfaces (UI) of the entry page and navigation bar of the controller.

The controller mode (Figure 6-14 (a)) works as a virtual joystick. The configuration parameters of the bending angle and direction angle (θ, φ) of each single section can be controlled individually (i.e., the target section is assigned by ④) by the two control sticks (① and ②) arranged in the bottom left/ right corner respectively. A velocity limitation bar ⑤ can set the step of angle changes $\Delta\theta$ and $\Delta\varphi$ when operating the control sticks when the numerical control instructions are indicated by ⑥. An emergency-stop button ③ is also set aside to avoid abnormal actions from the control panel. The real-time view from the head camera is also visible on this App, to make it easier for users to observe the working environment.



(a)



(b)

Figure 6-14 UIs of the remote controller: (a) the 'Controller' page. (b) the 'Navigation' page. List of main buttons: ① control stick for bending angle; ② control stick for direction angle; ③ emergency stop; ④ section selection bar; ⑤ velocity limitation bar; ⑥ numerical control instructions indicator; ⑦ navigation coordinate selecting window; ⑧ for forward/ backward button.

The navigation mode (Figure 6-14 (b)) is developed based on the navigation strategy proposed in Chapter 6.1 . This UI can send the instruction to guide the robot go forward or backward along the pre-planned path (button ⑧), and different from the robot main controller programmed in LabVIEW, the App provides an auxiliary option for shape adjustment during navigation.

This controller App was originally developed for the 8-section continuum robot (the MiRoR) for demonstrating its ability on working in nuclear facilities, while it has been generalized to be applicable for all the general continuum robots with less than 8

sections. The feasibility of this remote-control scheme has been demonstrated both on the MiRoR robot and the prototype robot successfully to control the robot configuration and navigate the robot into a target area.

6.6 Experiment and validation for the navigation strategy

6.6.1 An overview

This chapter demonstrates 2 experiments to validate the navigation strategy.

Table 7 System capabilities in navigation work.

The Robot System	The Self-carried Vision System	
- Closed-loop navigation along a pre-planned path. - In-process path adjustment.	**Head cam**	Environment interaction
	Body cam array	Section-by-section tip-positioning

The experiment setup is shown in Figure 6-15 (a). The 2-section robot is equipped with a head camera and 3 camera & marker arrays at the base of each section and the distal tip. 3 Vicon markers are also attached to the rings (camera arrays), which allows the off-the-shelf tracking device (VICON system) to record the trajectory of the robot in the word coordinate.

The first experiment aims to investigate the performance of closed-loop navigation under the led-by-the-tip strategy (Chapter 6.2 , Stage 1) in an open space (experiment setup: as shown in Figure 6-1), where the trajectory error, path error, and tip-following error will be compared (will be explained later in Figure 6-16).

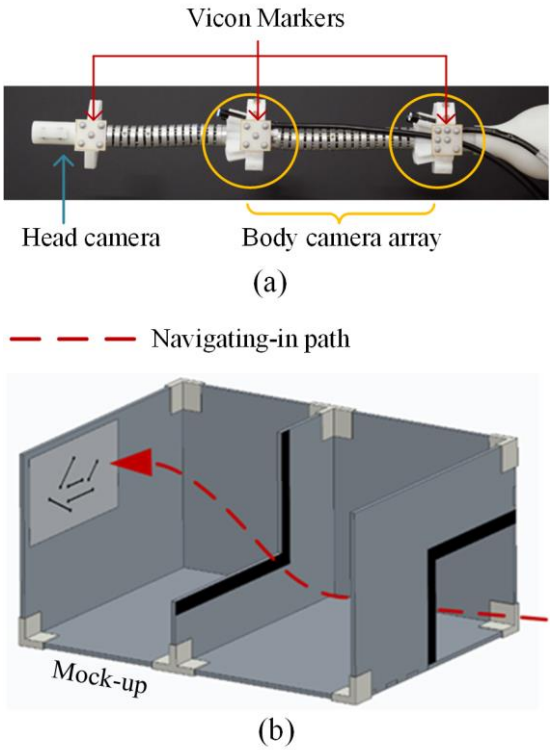


Figure 6-15 Experiment setup for navigation in a constructed environment. (a) the robot arm setup, the body camera arrays are attached at the section tips and the head camera is installed at the distal tip. (b) the mock-up for the demo, the red-dash line represents the pre-planned path to navigate the robot into the target area.

In the second experiment, the prototype robot is supposed to go through a mock-up along a pre-planned path (Figure 6-15 (b)). The mock-up has a set of barriers with identifiable boundaries and environmental markers (equivalent to obstacles), during the navigation, the capabilities of obstacle avoidance and online path adjustment (Chapter 6.3 , Stage 2) will be examined. After arriving at the target area, the robot is expected to measure the dimension of a target pattern attached to the mock-up (Chapter 6.4 , Stage 3, an example of the detection target is given in Figure 6-15 (b), at the end of the pre-planned path), and the measurement accuracy will be assessed. In both experiments,

6.6.2 Closed-loop navigation in an open space under the led-by-the-tip strategy

To implement the navigation strategy on the prototype robot, the algorithm of the led-by-the-tip strategy is programmed in both MATLAB and LabVIEW. The MATLAB

program functions in path pre-planning (Stage 1), while the LabVIEW program works for online path replanning (Stage 2), both of which use the same constraints and searching steps.

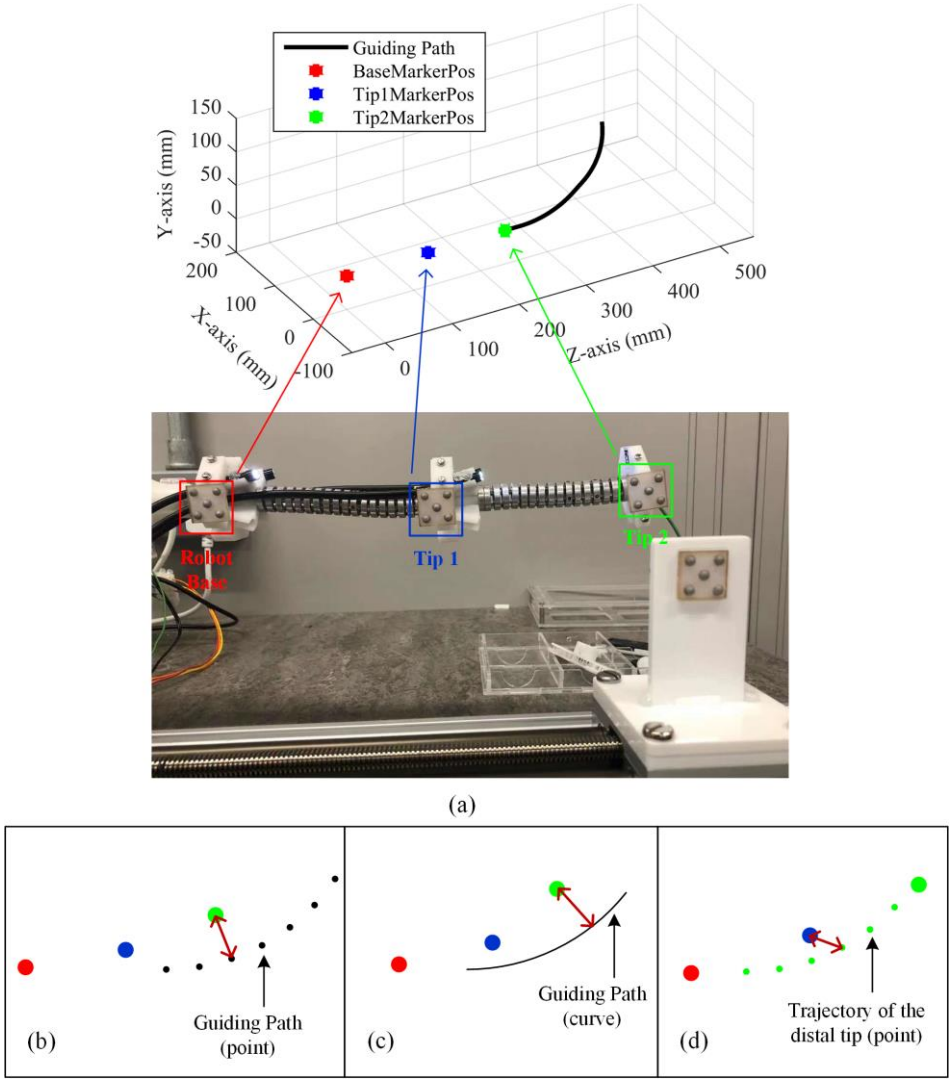


Figure 6-16 Graphic explanation of experiment results. (a) The instant VICON record of the robot base (red dot), the tip of a body section (Tip1, blue dot), and the distal tip (Tip 2, green dot) vs desired trajectories (pre-planned path). (b) Illustration of the path error (instant distal tip position vs discretized guiding path points). (c) Illustration of the trajectory error (instant distal tip position vs guiding path curve). (d) Illustration of the tip-following error (trajectory difference between the distal tip and the following tip(s) of body section(s)).

During the experiment, the robot goes forward along the pre-planned guiding path $[x_{tip}, y_{tip}, z_{tip}]_m$ (with corresponding guiding list for $[\theta_g, \varphi_g, Z_g]_m$) under open-loop control mode and closed-loop control mode respectively to compare the performance of

the navigation. The robot trajectories recorded by the VICON (Figure 6-16 (a)) system were analyzed to evaluate the led-by-the-tip strategy. The experiments were conducted in different quadrants (i.e., the robot is supposed to bend to different sides, and the pre-planned path & target destinations were directed in various directions). The performance was evaluated regarding path error, trajectory error, and tip-following error as demonstrated and explained in Figure 6-16.

More specifically, the path error indicates the point-to-point position error between the discretized guiding path and the distal tip position at each step Figure 6-16 (b). The trajectory error is illustrated in Figure 6-16 (c), which is the point-to-curve deviation, representing the minimum distance between the distal tip with the path curve at each step. Both the path error and the trajectory error are the indicators of trajectory accuracy of the distal tip in navigation. Finally, the tip-following error (Figure 6-16 (d)) is the indicator to assess the led-by-the-tip strategy, which is calculated by comparing the trajectory difference between the distal tip and the following body tip(s) (i.e., in the ideal case, the body tips should follow the trajectory of the distal tip).

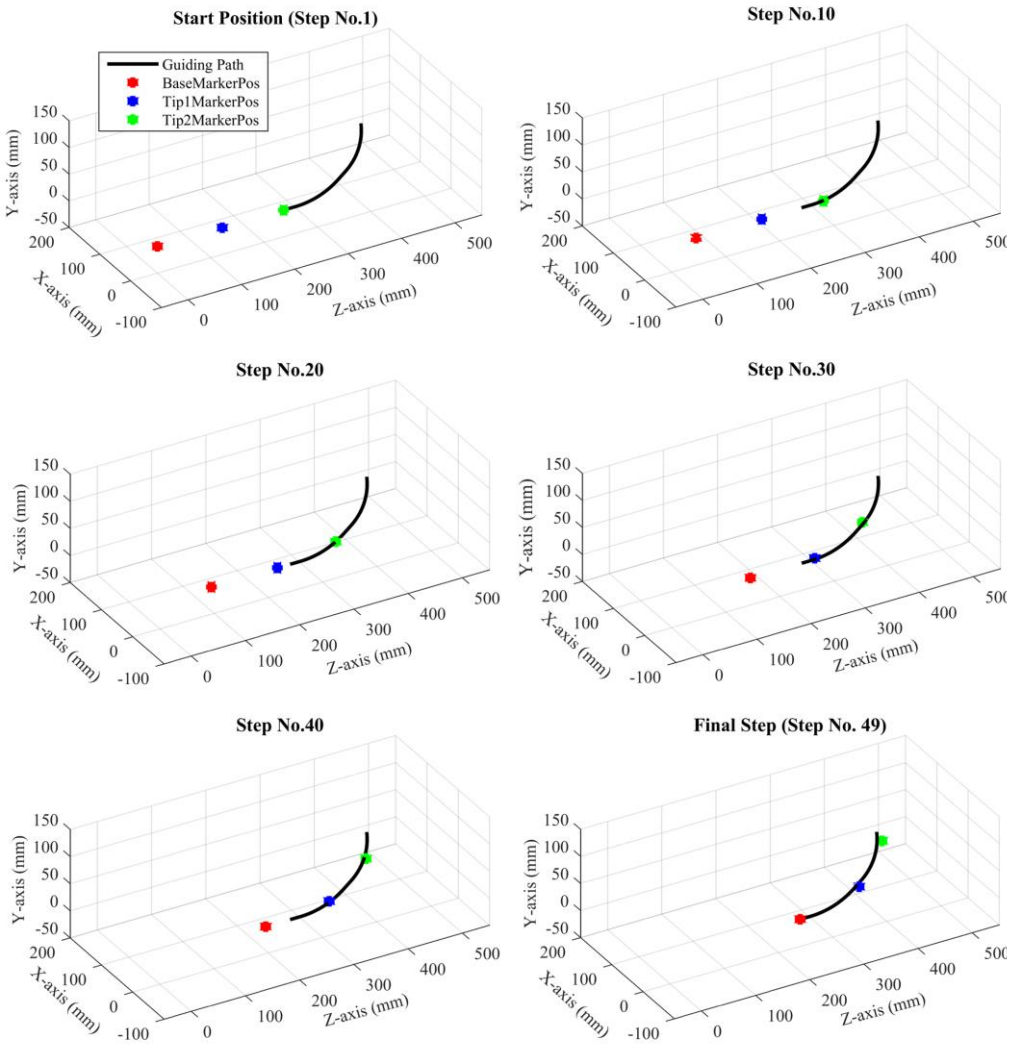


Figure 6-17 An example of a navigation process: the instant position recording of the robot tracked by VICON (red: robot base; blue: section tip; green: distal tip) against the desired robot trajectory (pre-planned guiding path, the solid black line).

Figure 6-17 is an example of the recorded navigation process (tracked by VICON) when the prototype robot snakes its way to go forward along a pre-planned guiding path (the black solid line), and the green, blue and red dots represent the instant position of the robot base, section tip, and distal tip respectively. It can be seen that the robot arm starts from a straight status and get on the track in sequence, where the distal tip follows the guiding path, and the section tip follows the distal tip before arriving at the target destination.

- **Navigation experiment results in the first quadrant of robot work-space (in the X-Y plane, left side)**:

In this test, both sections bend to the left side, and the desired destination of the distal tip is $[70.43, 84.70,\ 194.64]$ $(unit: mm)$ and the final robot configuration should be $\theta = \left[\frac{\pi}{6}, \frac{\pi}{4}\right]$; $\varphi = \left[\frac{\pi}{3}, \frac{\pi}{6}\right]$.

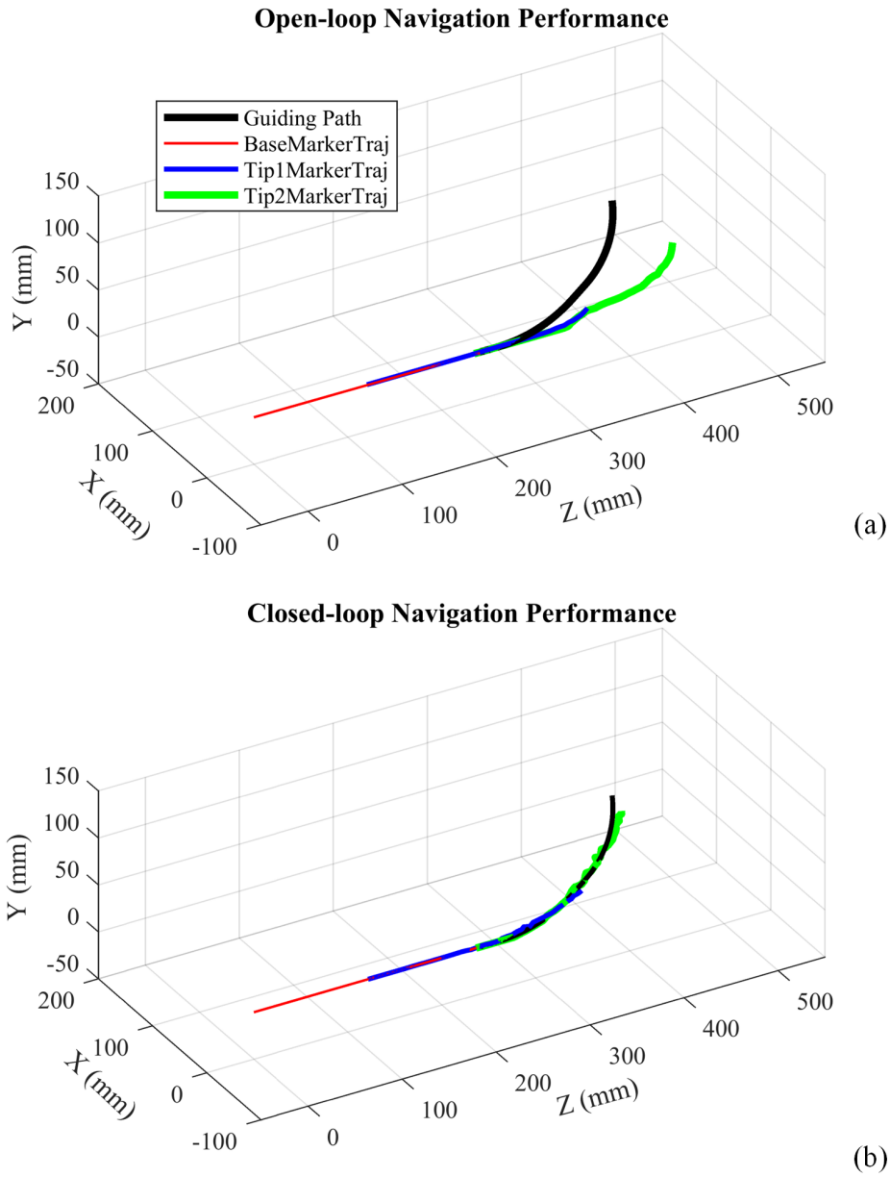


Figure 6-18 Robot trajectories recorded by VICON in the first-quadrant navigation experiment. (a) Under open-loop control mode. (b) Under closed-loop control mode.

Figure 6-18 compares the navigation performance under open-loop and closed-loop control, respectively. It can be observed that the under the closed-loop control aided by the hybrid sensing system (Chapter 4), the deviations between the pre-planned path and the trajectories of both tips are much smaller, and the controllability is significantly improved.

Then, a quantitative analysis is presented, regarding the three types of error along the path (i.e., the guiding path or guiding list has 45 steps to navigate the robot to the desired position, and the errors are presented in each step).

Table 8 Error comparison (quantitively) between open-loop and closed-loop control for the experiment in the first coordinate.

Mean error (mm)	Path Error	Trajectory Error	Tip-following Error
Open-loop control	27.74	26.25	1.49
Closed-loop control	5.40	3.56	2.51

Table 8 and Figure 6-19 illustrate the error distribution along the 49-step test route. According to the figures, it can be concluded:

- Both the path error and trajectory errors increase as the bending angle increases.
- Under open-loop control, the error is larger since the cable is not tensioned properly, especially when the errors accumulate as the step number increases.
- Since the cable is not distributed evenly (Figure 3-7), the robot can not be driven to the left side (in the first quadrant) properly by open-loop control. The open-loop control performance is much better in the second quadrant, which will be demonstrated later.
- The closed-loop controller significantly minimizes the path error and the trajectory error by 80.54% and 86.45% respectively.

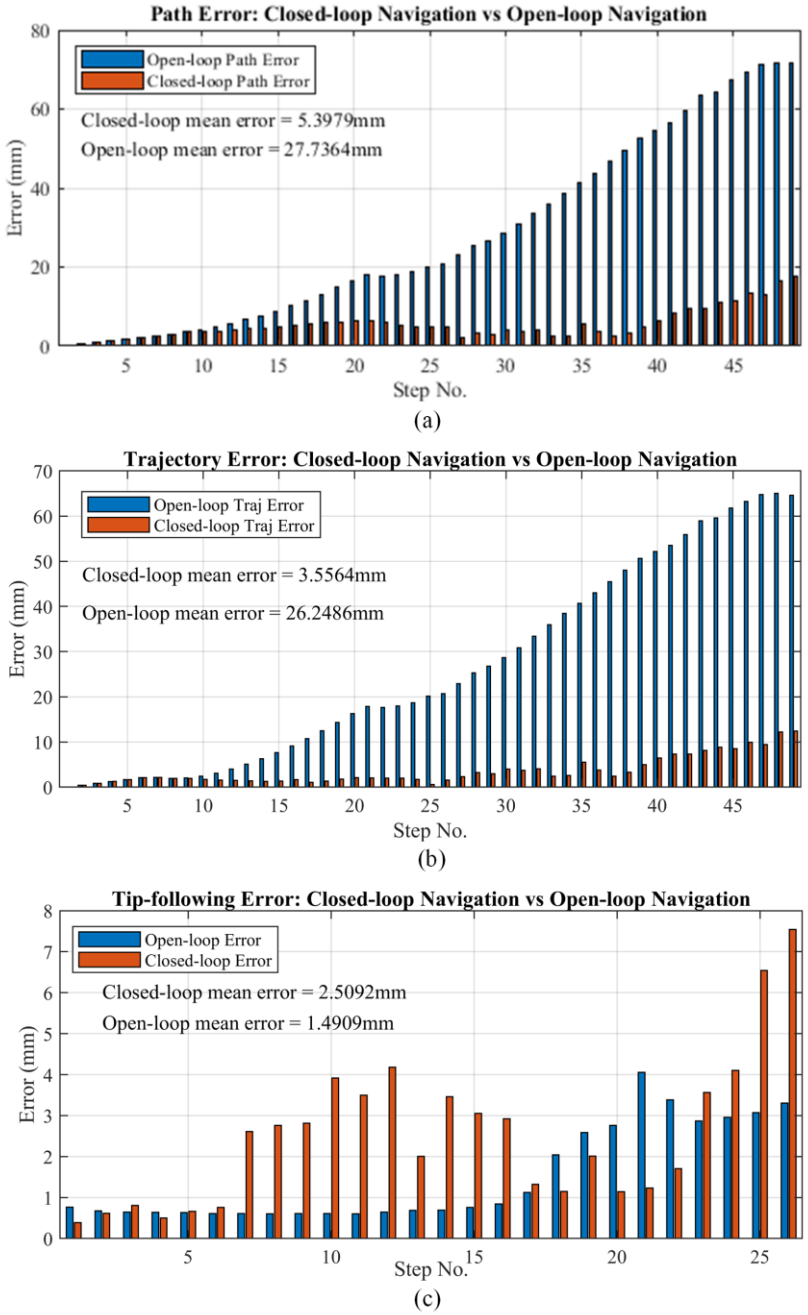


Figure 6-19 Error comparison (graphic) between open-loop navigation and closed-loop navigation in the first quadrant. (a) The path errors. (b) The trajectory errors. (c) The tip-following errors.

- The tip-following error is the indicator to evaluate the led-by-the-tip strategy, (i.e., to show how closely the section tips can follow the distal tip). The mean tip-following errors for open-loop control and closed-loop control are 1.49 mm and 2.51 mm respectively in the first-quadrant experiment. **However, in this**

specific case, the smaller tip-following error does not mean better performance.

- As mentioned before, since the improper cable tension under open-loop control (i.e., raised by uneven cable distribution), the robot does not achieve the desired pose and the bending angle is quite small when bent to the first quadrant. Thus, both the section tip and the distal tip only have minor movement, and the trajectories will be closer. Although the absolute open-loop tip-following error is smaller, the percentage error is still higher.

The pattern of tip-following error growth will be clearer in the second-quadrant experiment.

- **Navigation experiment results in the second quadrant of robot work-space (in the X-Y plane, right side)**:

In this test, both the sections bend to the right side, and the desired destination of the distal tip is $[-73.20, 68.74, 204.17]$ $(unit: mm)$, and the final robot configuration is $\theta = \left[\frac{\pi}{6}, \frac{2\pi}{9}\right]$; $\varphi = \left[\frac{2\pi}{3}, \pi\right]$.

Similar to the first-quadrant experiment, Figure 6-20 compares the navigation performance under open-loop and closed-loop control, smaller deviations can be found between the pre-planned path and the trajectories of both tips under closed-loop navigation, with better controllability.

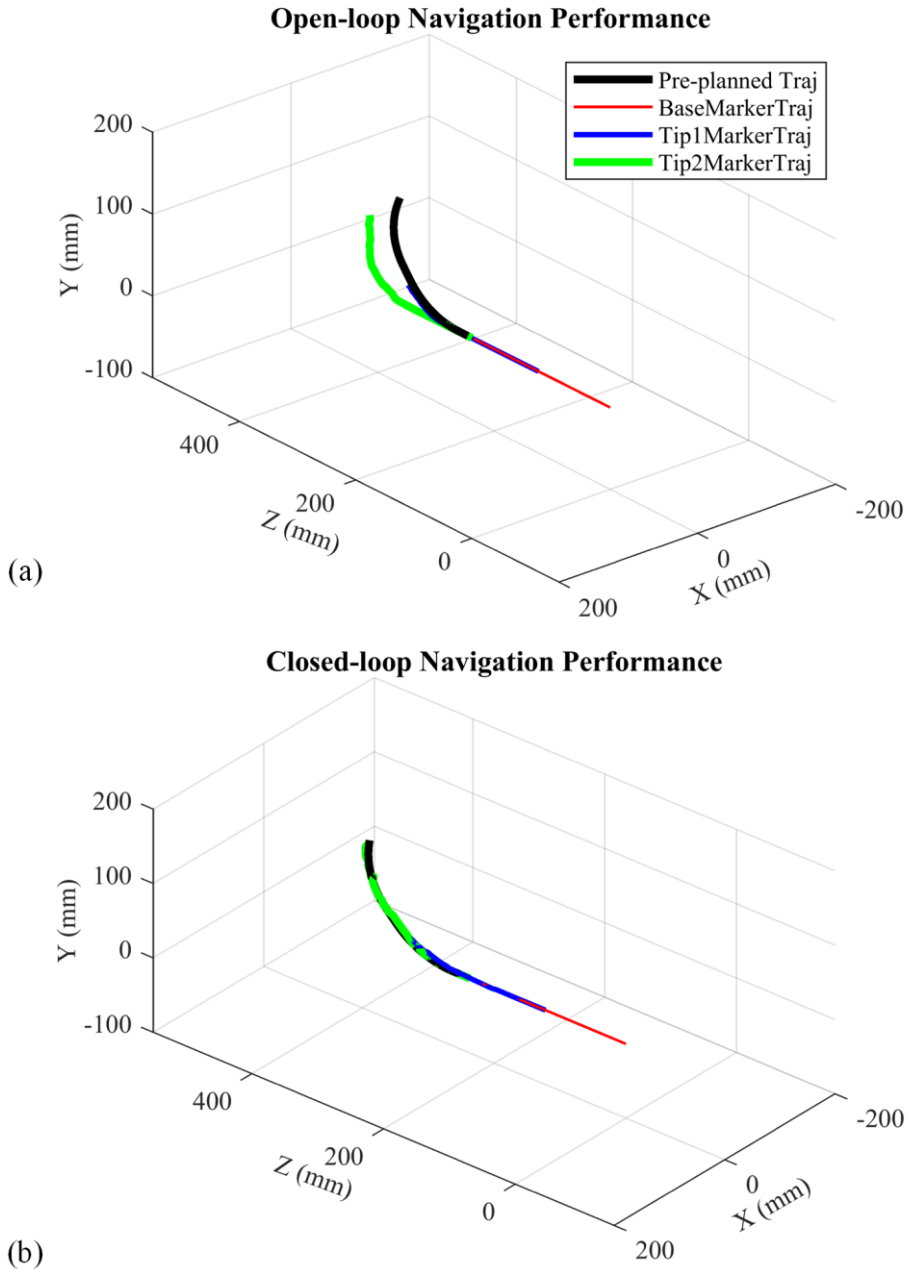


Figure 6-20 Robot trajectories recorded by VICON in the second-quadrant navigation experiment. (a) Under open-loop control mode. (b) Under closed-loop control mode.

A detailed quantitative error comparison is carried out in Table 9 and Figure 6-21.

Table 9 Error comparison (quantitively) between open-loop and closed-loop control for the experiment in the second coordinate.

Mean error (mm)	Path Error	Trajectory Error	Tip-following Error
Open-loop control	24.49	22.09	5.40
Closed-loop control	5.83	4.84	2.81

The following conclusions could be advised:

- The path error and trajectory error show similar trends with the experiment in the first quadrant, which are decreased by 76.19% and 78.01% respectively in closed-loop navigation.
- Regarding the tip-following error, the second-quadrant experiment shows a clearer pattern. In this case, since there are more cables distributed on the right side of the arm (Figure 3-7), the robot can step closer to the desired destination under open-loop control. It can be found that the tip-following error increases continuously as the robot goes forward.
- The closed-loop tip-following errors (mean value) are 2.51 and 2.81 mm in first & second-quadrant experiments respectively, which validates the stability of the led-by-the-tip strategy to some extent.

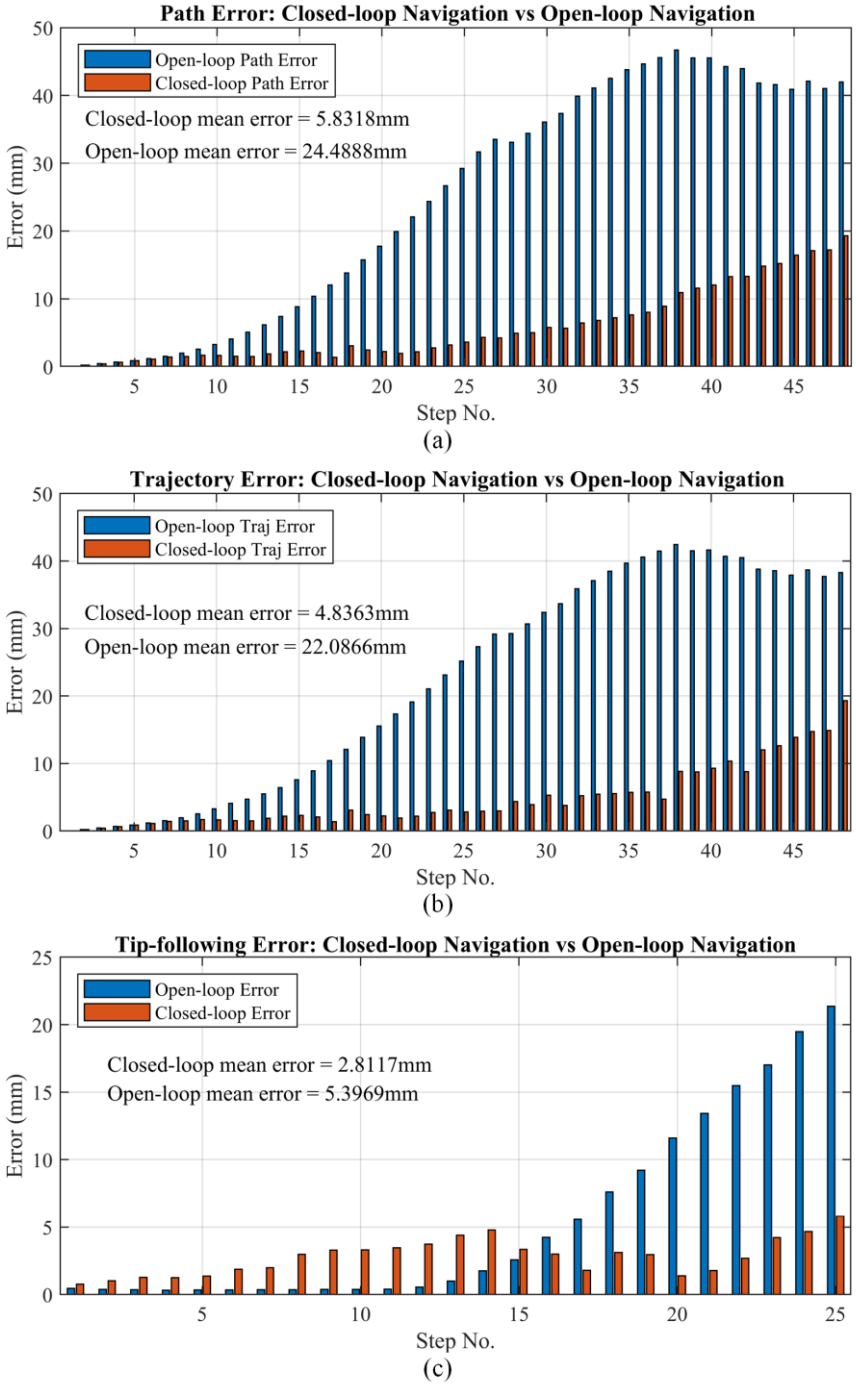


Figure 6-21 Error comparison (graphic) between open-loop control and closed-loop control (an experiment in the second coordinate). (a) The path errors. (b) The trajectory errors. (c) The tip-following errors.

6.6.3 Navigating into a mock-up of restricted spaces

As described in Chapter 6.3 , this experiment was designed to evaluate the performance of obstacle avoidance and online path adjustment, during closed-loop navigation.

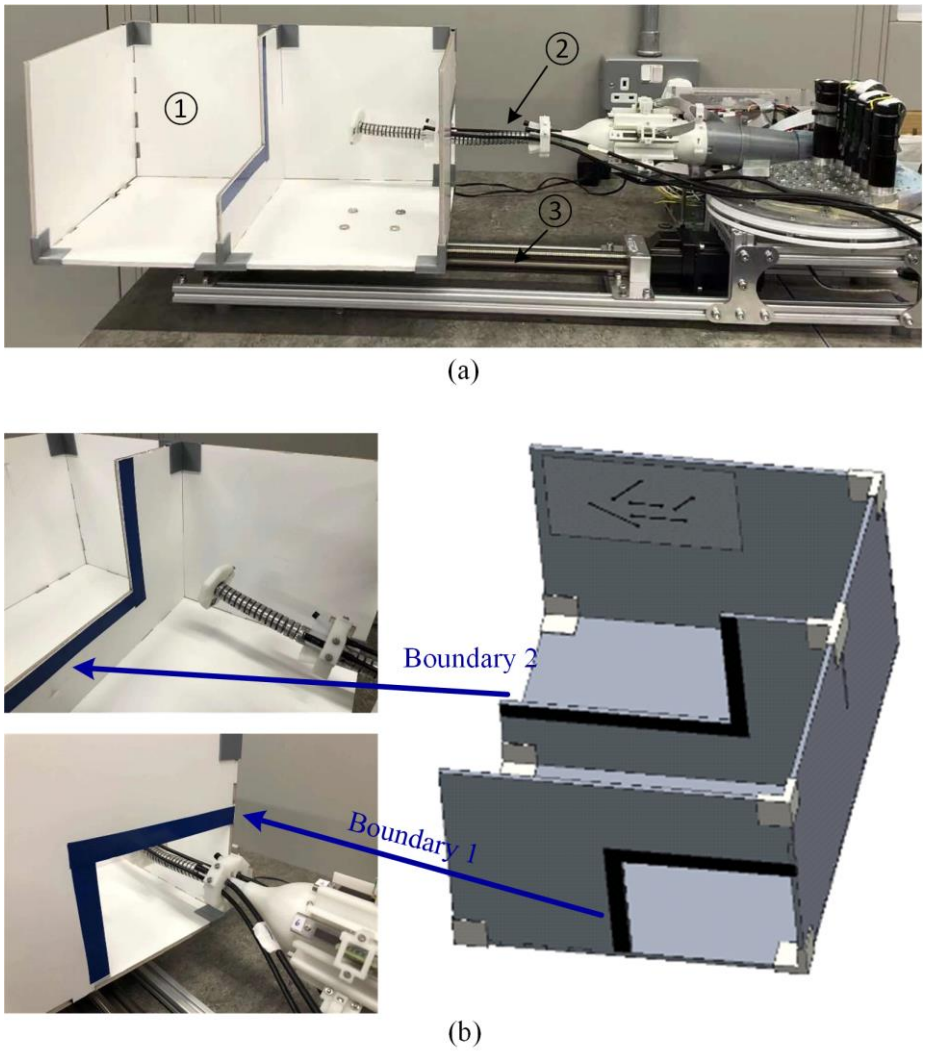


Figure 6-22 setups of the mock-up. (a) ① the mock-up; ② the continuum robot; ③ the linear rail. (b) the inner structure of the experiment mock-up.

Figure 6-22 shows the mock-up installed on a linear rail. The mock-up has 2 barriers, the continuum robot needs to pass the first boundary through the hole in the lower right corner and go forward while avoiding the obstruction on the second barrier.

According to the inner structure of the mock-up, a guiding path is computed first. This path is designed to let the robot just pass the boundaries, but there is a risk of collision for the attached camera & marker arrays. Figure 6-23 to Figure 6-25 compare the performance of the robot when passing through the second barrier under the open-loop control, closed-loop control, and closed-loop control with implemented obstacle avoidance algorithm respectively.



Figure 6-23 Performance of the robot under open-loop navigation when passing through the second barrier.

Under open-loop control, since the actuation cables are not tensioned properly, the robot was not fully bent to the desired configuration (i.e., according to the guiding list), thus it failed to pass the boundary and was stuck halfway.

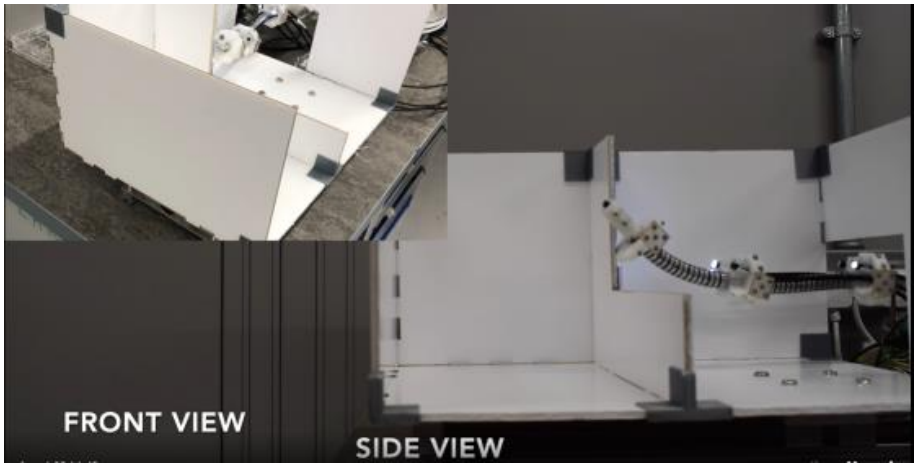


Figure 6-24 Performance of the robot under closed-loop navigation when passing through the second barrier.

In closed-loop navigation, the robot can fully bend with high tip-positioning accuracy. This time, the distal tip passes the boundary, but the head camera & marker array touched the boundary since it has a larger diameter (i.e., it is expected since when the path was planned, only the robot itself was in consideration). Thus, the robot was pushed up by the 'wall'.



Figure 6-25 Performance of the robot under closed-loop navigation with the implementation of obstacle avoidance algorithm when passing through the second barrier.

After implementing the algorithm of obstacle avoidance and online path adjustment, when an obstacle is detected, and its minimum distance with the pre-planned path is smaller than the desired safety distance, the controller will replan the following path. In this case, the distal tip and the head marker ring pass through the boundary successfully with a set safety distance (= 40 mm).

6.6.4 Object detection and measurement

Figure 6-26 demonstrates the experiment setup and the detected pattern in this test.

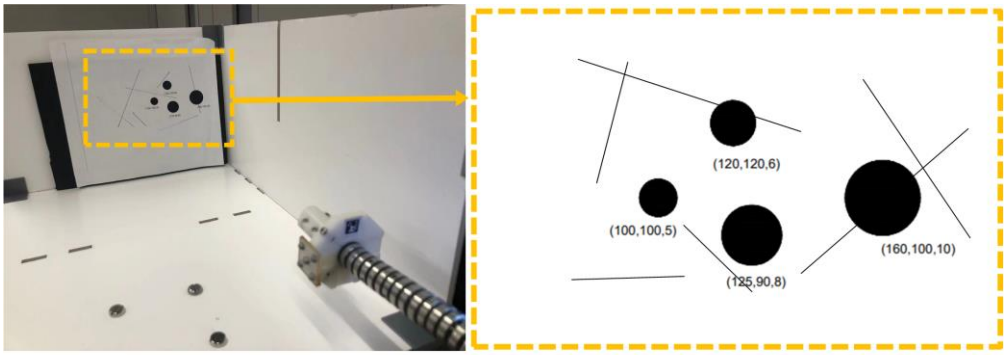


Figure 6-26 The experiment setup for object detection and measurement. (a) The robot system, the mock-up, and the detected pattern. (b) The detected pattern.

The objectives of this experiment are 1) to recognize these circular shapes from views of the head camera; 2) to measure the distance between the centroids of these circles.

Shape Detection for planner objects within the robot field-of-view

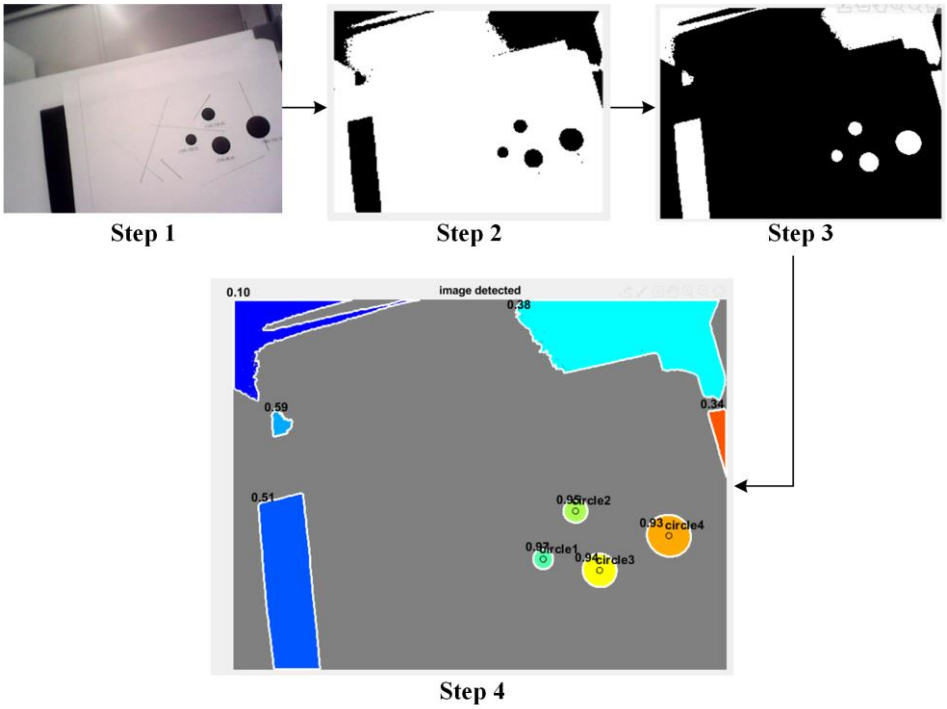


Figure 6-27 Procedure for shape detection: Step 1: read the raw image. Step 2: convert to a binary image. Step 3: noise reduction. Step 4: shape recognition.

Figure 6-27 shows the procedure to extract the target feature from an input image. Firstly, the raw image captured by the head camera is read by the processer and converted to a binary image, thus, the image data becomes a 1-layer matrix only containing a value of 0 and 255, which will be easier for the computer to split noisy points from the picture. Then, each data point is compared with its surroundings to figure out image noise (i.e., if the test data point has the opposite color from the surroundings, it will be treated as a noise point and deleted from the image, as demonstrated in Figure 6-27, Step 2 to Step 3). After all the color patches are distinguished, the boundary length and area of these patches can be counted. To select the desired shape(s), the criteria of area/ perimeter ratio is used:

$$r_{shape} = \frac{4\pi \times Area}{(perimeter)^2} \tag{47}$$

For instance, for a circle, the indicator ratio should be:

$$r_{circle} = \frac{4\pi \times (\pi r^2)}{(2\pi r)^2} = 1 \tag{48}$$

Thus, the threshold value can be set in a range of (0.85,1.15). All the shapes whose area/ perimeter ratio is in this range are selected (Figure 6-27, Step 4).

Distance measurement:

The distance measurement employs the method of linear triangulation (Chapter 6.4.2). The images captured by the head camera are firstly paired two-by-two (Figure 6-28 (a)), i.e., for an image set with n pictures, there are C_2^n (i.e., combinations) image pairs. Then, the two pictures within each image pair are matched by detected feature points, and the method of triangulation can compute the spatial coordinates of these matched feature points. In this experiment, the target features are the centers of detected circles. Finally, C_2^n results are output to take average.

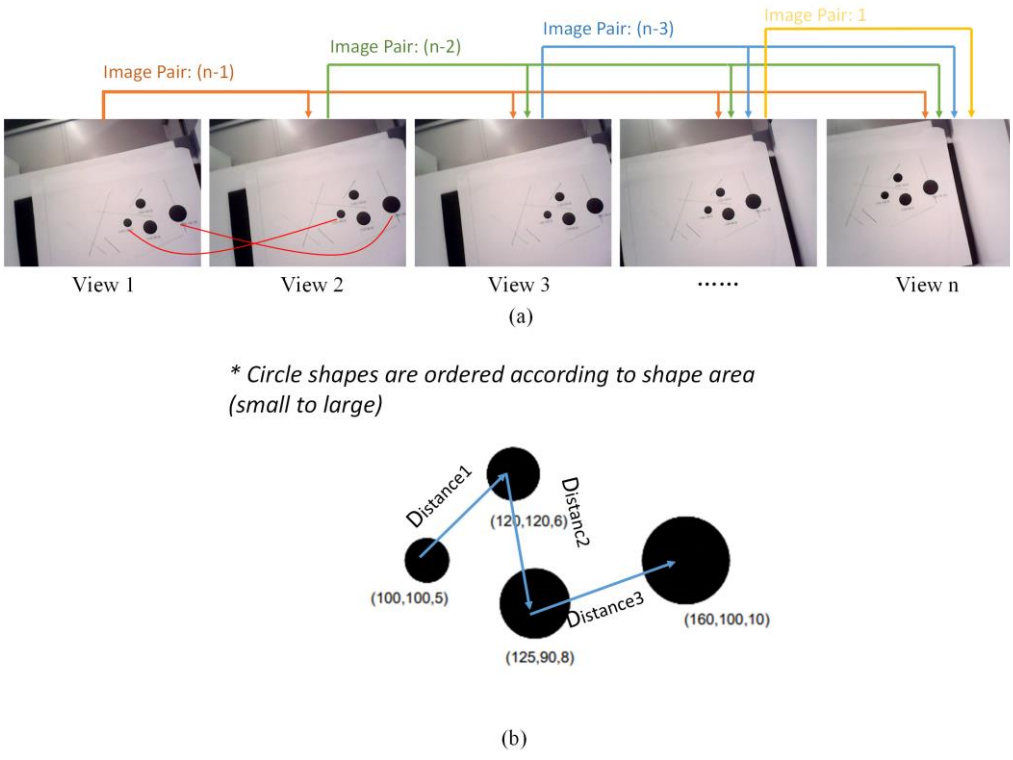


Figure 6-28 (a) Image matching. (b) the data measured in this experiment.

The experiment aims to measure the distance between the centroids of these circles (Distance1, Distance2, and Distance3 illustrated in Figure 6-28 (b)). There are 22

images of the objective pattern taken by the head camera, with 231 image pairs. Part of the measurement results is listed below:

Table 10 Distance measurement results from multiple image pairs (part).

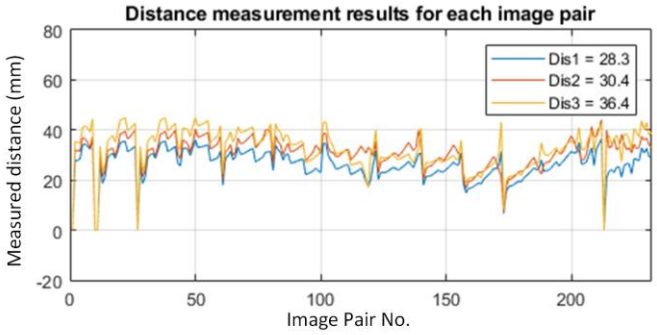
Image Pair No.	Distance 1	Distance 2	Distance 3
1	27.30	37.82	40.21
2	26.95	30.79	36.77
3	28.28	34.36	39.65
4	32.63	41.77	43.57
5	28.47	32.75	39.27
6	27.67	30.2381	36.70
7	29.02	33.0090	39.50
8	32.73	39.1966	43.78
9	30.02	32.0799	38.92
10	28.39	30.7218	36.80
11	29.79	32.5318	39.50
12	33.05	37.4179	44.00
……			
n-1	26.43	28.4761	33.88
n	29.21	32.5586	38.54
Mean	**29.28**	**33.84**	**39.36**

Table 11 Summary of the measurement results for object measurement.

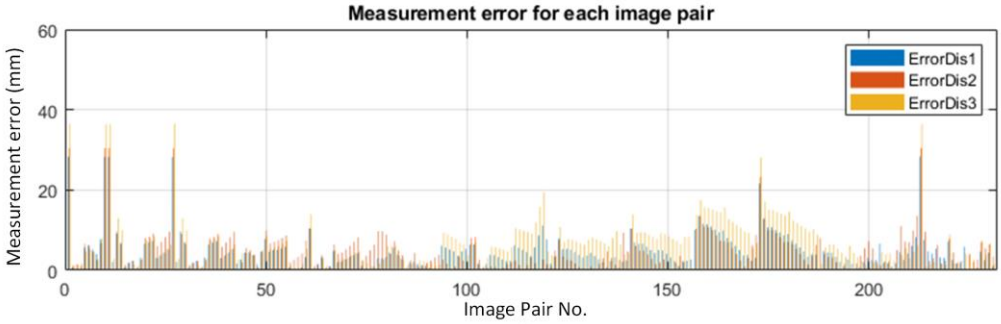
	Distance 1	Distance 2	Distance 3
Experimental (measured)	29.28	33.84	39.36
Theoretical (Actual)	28.28	30.41	36.40
Difference %	3.54%	11.28%	8.13%

Table 10 lists some of the measurement results calculated from the resultful image pairs, and

Table 11 presents the summary of the measurement results. It can be concluded that this method can provide a rough estimation of the 2-D distance information but has a slightly large data variance between the measurement results from different image pairs.



(a)



(b)

Image pairs No. vs measurement errors

28.3114	34.2615	35.0719	6.0000	13.0000	138.0000	0.0114
28.2843	31.5239	34.6265	1.0000	6.0000	6.0000	0.0157
28.3525	31.2058	35.2440	5.0000	3.0000	105.0000	0.0525
28.1824	32.5647	38.9910	23.0000	4.0000	367.0000	0.1176
28.1724	33.5216	35.2957	6.0000	12.0000	137.0000	0.1276
28.1204	29.7381	35.2247	5.0000	6.0000	108.0000	0.1796
28.5035	33.7673	36.9266	4.0000	18.0000	96.0000	0.2035
28.5632	35.3190	35.3124	5.0000	17.0000	119.0000	0.2632
28.0223	33.0118	35.9486	4.0000	17.0000	95.0000	0.2777
			.....			
17.5306	20.3733	21.3970	11.0000	3.0000	228.0000	10.7694
17.4591	20.4423	21.2515	10.0000	6.0000	213.0000	10.8409
17.1436	33.0931	17.0204	7.0000	20.0000	167.0000	11.1564
16.8225	19.6171	20.8795	10.0000	5.0000	212.0000	11.4775
16.7812	19.5563	20.8192	10.0000	4.0000	211.0000	11.5188
15.3899	17.5579	19.3166	11.0000	2.0000	227.0000	12.9101
14.8599	16.8772	18.9288	10.0000	3.0000	210.0000	13.4401
6.6521	7.1940	8.3179	11.0000	1.0000	226.0000	21.6479

Measurement results | No. first image | No. second image | Magnitude of abs(errors)

(c)

Figure 6-29 Graphic presentations of the results. (a) the measurement results for each image pair. (b) the measurement error for each image pair. (c) the measurement results sorted according to measurement error.

Figure 6-29 (a) and (b) draw the measurement results and the measurement error, followed by figure (c) compares the measurement error between different image pairs. It can be seen:

- Small movement of the robot tip (i.e., the orientation and position difference between the 2 viewpoints are small) always brings larger errors.
- The large measurement errors are always related to some specific images with inaccurate pose estimation, since the camera stereo parameters are actually unknown, but guessed from robot configuration parameters by the kinematics model, deleting these images can improve the measurement accuracy.

6.7 Chapter summary

This chapter presents a full adaptive navigation strategy for the prototype robot. Assisted by the closed-loop vision-based controller developed previously, the robot can be successfully navigated into a constructed confined space under the led-by-the-tip strategy, with low positioning error, trajectory error, and tip-following error (i.e., error within a degree of milometers). It's the first time to achieve semi-online path planning of a continuum robot under led-by-the-tip motion. Furthermore, based on the tip-attached camera and robot kinematics, a pseudo-triangulation method is also proposed, which equips the prototype robot with the outward-sensing ability. In this way, the robot can interact with the working environment to execute more complex tasks such as object measurement, obstacle detection & avoidance, etc. Further work could consider developing this navigation work frame under unstructured environments.

Chapter 7 Conclusion and further work

This chapter reviews the objectives of this research work and summarizes the overall outcomes and achievements presented in this thesis, discussing how the problems are solved by the current robot system. Finally, some advice is given in terms of the research potential and further work in the field of continuum robots.

7.1 Academic finding and engineering validation

As discussed in Chapter 1 , this Ph.D. project aims to propose a compositive robot sensing and control scheme, considering all the possible error sources on a cable-driven continuum robot with twin-pivot arm design and rotary actuation system design (i.e., the tendons are driven by the motor-spool structure). To sum up, errors are mainly existing in cable displacement control and modeling. Except for the inherent problem of cable elongation caused by inhomogeneous friction distributed in the arm structure, the cable displacement control errors are also introduced by the rotary actuation design, including backlash of the motor and unideal winding of the cable (on the spool). Regarding modeling, the conventional kinematics are established under ideal working conditions, which ignores the effects of self-weight, coupling, etc., while they are nonnegligible in practical operation. To improve the sensibility, controllability, and practicability of a cable-driven continuum robot system, this research emphasizes sensing solutions, feedback controller design, and exploring more application scenarios such as navigation in an obstructed environment and remote control. The following subsections will explicitly present these achievements.

7.1.1 Novel hybrid sensing solution developed to evaluate and feedback system errors of continuum robots

In Chapter 4 , a comprehensive hybrid sensing system is presented to achieve closed-loop control both in the joint space (i.e., cable displacement error) and the workspace

(i.e., tip positioning error). The hybrid sensing solution consists of a dual displacement sensor system and a self-carried visual tracking system. The former works in joint space, employing the concept of a secondary displacement sensor (rotary & linear displacement encoders) to monitor and compensate for the multi-source cable displacement control error. The visual system duties in tasks space to track the tip position of the robot section-by-section. Significantly, it is the first time that an embedded camera device is used to indicate the robot's own configuration parameters.

- The dual displacement sensor system consists of rotary encoders installed at the end of actuation motors (which indicates theoretical cable displacement calculated from the kinematics model) and a group of linear encoders mounted between the arm structure and actuation pack (which shows the actual cable displacement at the proximal end of the robot arm). For each actuation cable, the idea is to compare the readings from the two types of encoders. In this way, the multi-source displacement error caused by unideal winding, motor backlash, and cable elongation within the actuation pack can be evaluated.
- The self-carried visual tracking system is grouped by some serially arranged observe-reference camera & marker arrays equipped along the arm structure. The working principle is to install some miniatured cameras on the base of each section and attach some ArUco markers at the tips. After calibration, tip-attached markers can be tracked in each reference coordinate. In this way, a compact configuration sensing solution is carried out with guaranteed feedback accuracy, which makes it possible to apply in a confined space. It has been validated that for a single section arm, the vision system can offer section tip-tracking with a mean error of 0.67 mm in its workspace.

7.1.2 The closed-loop controller developed for a multiple-section continuum robot (a model-less control strategy)

Following the concept of hybrid sensing, a two-level closed-loop controller is developed, working to compensate for the measured control error in the joint space and workspace. In the joint space, a two-stage cable displacement control approach is proposed, working with the dual displacement sensor system, which evaluates the cable displacement error first and then compensates for the measured error in a secondary control. In the workspace, an indirect measurement based tip-adjustment strategy is designed. Essentially, it is a model-less control method that approaches the robot tip gradually to the desired position according to feedback error in cartesian coordinates, whose advantage is that it avoids the multiple-solution problem of inverse kinematics. The experiment results demonstrate the performance of such a controller regarding preventing cable slack, minimizing open-loop control error, and increasing end-effector control accuracy in both local & global coordinates.

- The main idea of the two-stage cable displacement control approach is to divide the actuation cables into driving cables (whose displacements are negative) and passive cables, and then use different strategies to control the two groups of cables. In this way, the driving cables can be properly tensioned. When following a pre-planned path repeatedly, compared with the conventional rotary actuation mechanism, the maximum positioning error of the robot tip decreases from 9.96 mm to 6.70 mm, and the mean error decreases from 5.79 mm to 3.13 mm in the x-y plane.
- The experiment results validate the performance of the proposed controller: 1) the average tip-positioning error (for the end-effector) decreases from 6.50 mm to 1.72 mm in the world coordinate; 2) the local tip positioning error in the reference coordinate for both the tip section and base section is converged to the user-input desired value (in this case within 0.8 mm).

7.1.3 An adaptive navigation strategy developed to conduct the robot enter more complex working scenarios

A semi-online navigation strategy is put forward to enhance the practicability of such a robot. This navigation strategy aims to guide a slender robot pass through some working environments with roadblocks. The core idea of this method is the led-by-the-tip strategy, under such a work frame, a pre-planned path is generated according to the environment feature, and the distal tip of a continuum robot would follow the pre-planned path to go forward while the robot body follows the trajectory of the distal tip. The tip-attached camera works to detect potential obstacles, with the online path adjustment approach, the robot can interact with the environment and 'make the decision' for its next move. Furthermore, based on the tip-attached camera and robot kinematics, a pseudo-triangulation method is also proposed, which equips the prototype robot with the outward-sensing ability. In this way, the robot can interact with the working environment to execute more complex tasks such as object measurement, obstacle detection & avoidance, etc.

- Two sets of experiments are conducted to test the feasibility of this method. Firstly, the vision-aided closed-loop navigation strategy is tested in an open space, it demonstrates high accuracy of path-following (mean error of 5.40 and 4.68 mm in the first/ second coordinate) and tip-following (mean error of 2.51 and 2.81 mm in the first/ second coordinate). Secondly, the robot is navigated into a mock-up with barriers, the obstacle avoidance strategy does work to let the robot change its way and arrive at the target area.
- It's the first time to achieve semi-online path planning of a continuum robot under led-by-the-tip motion.
- A remote controller based on Android Studio is also developed, for the safe operation of continuum robots in extreme working environments such as nuclear facilities.

7.2 Further work

This thesis proposes a sensing system designed for continuum robot systems with a derived closed-loop controller and navigation strategy, which has been validated on a 2-section prototype robot as an example of a multiple-section application. However, some difficulties and challenges still occurred, bringing potential research opportunities. This chapter identifies these problems and gives out some suggestions for further work.

7.2.1 Sensing solutions for continuum robots

- **Tactile sensor and compliant control**

  The tactile sensor can equip the robot with a sense of touch. It can be mounted on the distal tip of the continuum robot or wrapped around the arm structure (to the side of the arm) to let the robot 'feel' the working environment and avoid a collision. This kind of sensor is an applicable substitution for 3-D shaping sensors for the passive sections of a continuum robot (i.e., non-actuated sections) or in applications with low control accuracy required (e.g., go through a channel).

- **FBG sensor for precise 3D shape sensing**

  FBG is the abbreviation of Fibre Bragg Gratings, which is a kind of optical sensor (as introduced in Chapter 2). It is an ideal solution for 3-D shape sensing and structure reconstruction with high accuracy and fast response speed, which can provide feasible feedback for robot configuration and tip position. Also, the small-diameter FBG fibers can be arranged in various patterns and imbedded into soft materials like rubber, meeting customer scenarios and requirements (e.g., can be used to develop a shape-sensing sleeve for the continuum robot). The only consideration is the high cost of this sensor and its processor.

- **Embedded shape-sensing material**

  Using shape-sensing material to build the continuum robots or implanting such material into the structure of a robot is another solution. This kind of material

always works to detect the local strain and restore the shape by the method of integration. Some examples of applicable candidates are electro-conductive yarn, ionic liquid, optical fibers, optoelectronic sensors, etc.

7.2.2 Robot controller design

- **Learning based control**

  Learning-based methods such as ANN and RNN are another trend for robot controller design. As introduced in Chapter 2 , it always works to optimize the coefficient of conventional PID controllers or propagate the non-linear models. This kind of controller has great potential in applying to continuum robots to predict errors or build inverse kinematics models.

- **Hybrid control strategy**

  Any single control method has its disadvantages, a hybrid control strategy tries to integrate the advantages of different controllers and avoid the deficiencies. For example, the closed-loop controller developed in this thesis can be combined with a learning-based controller. The leaning-based controller can collect data when the robot is operated (at the incipient stage) and propagate its model online. Thus, the time-consuming (off-line) pre-training process can be avoided, which may also improve the efficiency of the closed-loop controller for tip error compensation.

- **Any control method with online optimization**

  As a non-linear system, the continuum robot system has several time-varying and coordinate-varying control uncertainties during operation. Thus, online optimization might be a solution to predict the ‘local models’ for a robot. For example, in model predictive control, the model is continuously updated according to the control output.

7.2.3 Further development on the vision-based sensing system and environment interactivity

- **Multi-perspective camera-array design**

  With the development of miniature camera devices, more possibilities can be explored regarding the design of camera arrays. For example, the chip cameras (camera modules) can be employed and attached to the side of a robot, working to provide more observation perspectives of the working environment.

- **Integrated vision processer**

  As the number of cameras increases, an integrated vision processor is required to control all the cameras and store & process the images. In the current system, the MATLAB Supports Package for USB Webcams works as the camera controller, where the images from different cameras are captured in sequence. However, this method is only applicable in low-speed applications with a limited number of devices, the synchronization of image acquisition cannot be guaranteed. Furthermore, the serial program of image processing also delays the response speed of the vision system when more cameras are added, which can be solved by using a parallel processor (hardware or software).

- **Implementation of advanced computer vision algorithms**

  As the control accuracy of the robot improved and more information is extracted from the working environment, it is possible to develop more advanced visual applications such as 3-D reconstruction, augmented reality, etc. There are some open-source libraries that can be compiled and implemented with the vision system like Open-slam and OpenCV etc.

- **Orientation error evaluation according to the method of robot-hand-eye-calibration**

  In this research, the proposed self-carried vision system employs the position measurements of attached visual markers to track the tip positions of a continuum robot section-by-section. The measurements of orientation are

rejected for the large deviation near the singularity point. As the vision devices developed, if images with higher quality can be extracted from the working environment, the orientation error of robot tips can be evaluated by the method of hand-eye-calibration, which is a mature approach to locate the robot against the working environment, thus the orientation error of the robot tip can be evaluated by observing the surrounding features.

REFERENCE LIST

[1] B. Chen, J. Wan, L. Shu, P. Li, M. Mukherjee, and B. Yin, "Smart factory of industry 4.0: Key technologies, application case, and challenges," *Ieee Access*, vol. 6, pp. 6505-6519, 2017.

[2] Y. Yin, K. E. Stecke, and D. Li, "The evolution of production systems from Industry 2.0 through Industry 4.0," *International Journal of Production Research*, vol. 56, no. 1-2, pp. 848-861, 2018.

[3] D. Imkamp *et al.*, "Challenges and trends in manufacturing measurement technology–the "Industrie 4.0" concept," *Journal of sensors and sensor systems*, vol. 5, no. 2, pp. 325-335, 2016.

[4] E. I. S. UK Gov Department for Business. "Regulation for the Fourth Industrial Revolution." https://www.gov.uk/government/publications/regulation-for-the-fourth-industrial-revolution/regulation-for-the-fourth-industrial-revolution (accessed 11/09, 2022).

[5] I. Bonev. "Delta parallel robot-the story of success." www.parallelmic.org (accessed 03/04, 2023).

[6] Igus. "SCARA robot - fast, flexible and configurable." https://www.igus.co.uk/info/n21-drylin-scara-robot (accessed 03/04, 2023).

[7] Igus. "What is a cartesian robot and how is it used in automation?" https://toolbox.igus.com/motion-plastics-blog/cartesian-robots-for-automation (accessed 03/04, 2023).

[8] ABB. "Articulated robots." https://new.abb.com/products/robotics/robots/articulated-robots (accessed 03/04, 2023).

[9] X. Dong *et al.*, "Development of a slender continuum robotic system for on-wing inspection/repair of gas turbine engines," *Robotics and computer-integrated manufacturing*, vol. 44, pp. 218-229, 2017.

[10] P. K. Singh and C. M. Krishna, "Continuum arm robotic manipulator: A review," *Universal Journal of Mechanical Engineering*, vol. 2, no. 6, pp. 193-198, 2014.

[11] D. Axinte *et al.*, "MiRoR—Miniaturized robotic systems for holistic in-situ repair and maintenance works in restrained and hazardous environments," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 2, pp. 978-981, 2018.

[12] C. Shi *et al.*, "Shape sensing techniques for continuum robots in minimally invasive surgery: A survey," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 8, pp. 1665-1678, 2016.

[13] M. Wang, D. Palmer, X. Dong, D. Alatorre, D. Axinte, and A. Norton, "Design and development of a slender dual-structure continuum robot for in-situ aeroengine repair," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018: IEEE, pp. 5648-5653.

[14] F. Qi, F. Ju, D. Bai, Y. Wang, and B. Chen, "Motion modelling and error compensation of a cable‐driven continuum robot for applications to minimally invasive surgery," *The international journal of medical robotics and computer assisted surgery*, vol. 14, no. 6, p. e1932, 2018.

[15] X. Dong, M. Raffles, S. Cobos-Guzman, D. Axinte, and J. Kell, "A novel continuum robot using twin-pivot compliant joints: design, modeling, and validation," *Journal of Mechanisms and Robotics*, vol. 8, no. 2, 2016.

[16] W. Ba, X. Dong, A. Mohammad, M. Wang, D. Axinte, and A. Norton, "Design and validation of a novel fuzzy-logic-based static feedback controller for tendon-driven continuum robots," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 6, pp. 3010-3021, 2021.

[17] N. Simaan *et al.*, "Design and integration of a telerobotic system for minimally invasive surgery of the throat," *The International journal of robotics research*, vol. 28, no. 9, pp. 1134-1153, 2009.

[18] I. D. Walker, "Continuous backbone "continuum" robot manipulators," *International Scholarly Research Notices*, vol. 2013, 2013.

[19] T. George Thuruthel, Y. Ansari, E. Falotico, and C. Laschi, "Control strategies for soft robotic manipulators: A survey," *Soft robotics*, vol. 5, no. 2, pp. 149-163, 2018.

[20] W. S. Rone and P. Ben-Tzvi, "Continuum robotic tail loading analysis for mobile robot stabilization and maneuvering," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2014, vol. 46360: American Society of Mechanical Engineers, p. V05AT08A009.

[21] A. L. Orekhov, V. A. Aloi, and D. C. Rucker, "Modeling parallel continuum robots with general intermediate constraints," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017: IEEE, pp. 6142-6149.

[22] K. Xu and N. Simaan, "Actuation compensation for flexible surgical snake-like robots with redundant remote actuation," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006: IEEE, pp. 4148-4154.

[23] B. L. Conrad and M. R. Zinn, "Interleaved continuum-rigid manipulation approach: Development and functional evaluation of a clinical scale manipulator," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014: IEEE, pp. 4290-4296.

[24] M. Li, R. Kang, S. Geng, and E. Guglielmino, "Design and control of a tendon-driven continuum robot," *Transactions of the Institute of Measurement and Control*, vol. 40, no. 11, pp. 3263-3272, 2018.

[25] K. Cao, R. Kang, D. T. Branson III, S. Geng, Z. Song, and J. S. Dai, "Workspace analysis of tendon-driven continuum robots based on mechanical interference identification," *Journal of Mechanical Design*, vol. 139, no. 6, p. 062303, 2017.

[26] B. Bardou, P. Zanne, F. Nageotte, and M. de Mathelin, "Control of a multiple sections flexible endoscopic system," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010: IEEE, pp. 2345-2350.

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

[28] A. Yeshmukhametov and K. Koganezawa, "Design and Development of Wire-Driven Pulley Guided Continuum Robot (WPGCR) Arm and Friction Analysis."

[29] W. McMahan *et al.*, "Field trials and testing of the OctArm continuum manipulator," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006: IEEE, pp. 2336-2341.

[30] V. C. Anderson and R. C. Horn, "Tensor arm manipulator," ed: Google Patents, 1970.

[31] I. D. Walker and M. W. Hannan, "A novel'elephant's trunk'robot," in *1999 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (Cat. No. 99TH8399)*, 1999: IEEE, pp. 410-415.

[32] Z. Li and R. Du, "Design and implementation of a biomimetic wire-driven underactuated serpentine manipulator," *Transaction on Control and Mechanical Systems*, vol. 1, pp. 250-258, 2012.

[33] J. S. Mehling, M. A. Diftler, M. Chu, and M. Valvo, "A minimally invasive tendril robot for in-space inspection," in *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006. BioRob 2006.*, 2006: IEEE, pp. 690-695.

[34] D.-G. Choi, B.-J. Yi, and W.-K. Kim, "Design of a spring backbone micro endoscope," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007: IEEE, pp. 1815-1821.

[35] H.-S. Yoon and B.-J. Yi, "A 4-DOF flexible continuum robot using a spring backbone," in *2009 International Conference on Mechatronics and Automation*, 2009: IEEE, pp. 1249-1254.

[36] C. G. Frazelle, A. Kapadia, and I. Walker, "Developing a kinematically similar master device for extensible continuum robot manipulators," *Journal of Mechanisms and Robotics*, vol. 10, no. 2, p. 025005, 2018.

[37] M. C. Yip and D. B. Camarillo, "Model-less feedback control of continuum manipulators in constrained environments," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 880-889, 2014.

[38] I. A. Gravagne, C. D. Rahn, and I. D. Walker, "Large deflection dynamics and control for planar continuum robots," *IEEE/ASME transactions on mechatronics*, vol. 8, no. 2, pp. 299-307, 2003.

[39] A. Bajo, R. E. Goldman, and N. Simaan, "Configuration and joint feedback for enhanced performance of multi-segment continuum robots," in *2011 IEEE International Conference on Robotics and Automation*, 2011: IEEE, pp. 2905-2912.

[40] M. W. Hannan and I. D. Walker, "Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots," *Journal of robotic systems*, vol. 20, no. 2, pp. 45-63, 2003.

[41] P. Dario *et al.*, "A novel mechatronic tool for computer-assisted arthroscopy," *IEEE transactions on information technology in biomedicine*, vol. 4, no. 1, pp. 15-29, 2000.

[42] M. Cianchetti, A. Arienti, M. Follador, B. Mazzolai, P. Dario, and C. Laschi, "Design concept and validation of a robotic arm inspired by the octopus," *Materials Science and Engineering: C*, vol. 31, no. 6, pp. 1230-1239, 2011.

[43] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi, "Learning the inverse kinetics of an octopus-like manipulator in three-dimensional space," *Bioinspiration & biomimetics*, vol. 10, no. 3, p. 035006, 2015.

[44] S. Sanan, J. Moidel, and C. G. Atkeson, "A continuum approach to safe robots for physical human interaction," in *International Symposium on Quality of Life Technology*, 2011: Citeseer.

[45] Y. Zhao *et al.*, "A soft continuum robot, with a large variable-stiffness range, based on jamming," *Bioinspiration & biomimetics*, vol. 14, no. 6, p. 066007, 2019.

[46] Y. Li, T. Ren, Y. Chen, and M. Z. Chen, "A variable stiffness soft continuum robot based on pre-charged air, particle jamming, and origami," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020: IEEE, pp. 5869-5875.

[47] L. Tang, L. Zhu, X. Zhu, and G. Gu, "Confined spaces path following for cable-driven snake robots with prediction lookup and interpolation algorithms," *Science China Technological Sciences*, vol. 63, no. 2, pp. 255-264, 2020.

[48] L. Tang, J. Wang, Y. Zheng, G. Gu, L. Zhu, and X. Zhu, "Design of a cable-driven hyper-redundant robot with experimental validation," *International Journal of Advanced Robotic Systems*, vol. 14, no. 5, p. 1729881417734458, 2017.

[49] L. Tang, J. Huang, L.-M. Zhu, X. Zhu, and G. Gu, "Path tracking of a cable-driven snake robot with a two-level motion planning method," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 935-946, 2019.

[50] E. Amanov, J. Granna, and J. Burgner-Kahrs, "Toward improving path following motion: hybrid continuum robot design," in *2017 IEEE international conference on robotics and automation (ICRA)*, 2017: IEEE, pp. 4666-4672.

[51] R. H. Gaylord, "Fluid actuated motor system and stroking device," ed: Google Patents, 1958.

[52] D. Trivedi, A. Lotfi, and C. D. Rahn, "Geometrically exact models for soft robotic manipulators," *IEEE Transactions on Robotics,* vol. 24, no. 4, pp. 773-780, 2008.

[53] A. D. Kapadia, K. E. Fry, and I. D. Walker, "Empirical investigation of closed-loop control of extensible continuum manipulators," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014: IEEE, pp. 329-335.

[54] D. Braganza, D. M. Dawson, I. D. Walker, and N. Nath, "A neural network controller for continuum robots," *IEEE transactions on robotics,* vol. 23, no. 6, pp. 1270-1277, 2007.

[55] M. Cianchetti, T. Ranzani, G. Gerboni, I. De Falco, C. Laschi, and A. Menciassi, "STIFF-FLOP surgical manipulator: Mechanical design and experimental characterization of the single module," in *2013 IEEE/RSJ international conference on intelligent robots and systems*, 2013: IEEE, pp. 3576-3581.

[56] M. Rolf and J. J. Steil, "Efficient exploratory learning of inverse kinematics on a bionic elephant trunk," *IEEE transactions on neural networks and learning systems,* vol. 25, no. 6, pp. 1147-1160, 2013.

[57] T. Mahl, A. Hildebrandt, and O. Sawodny, "A variable curvature continuum kinematics for kinematic control of the bionic handling assistant," *IEEE transactions on robotics,* vol. 30, no. 4, pp. 935-949, 2014.

[58] A. Melingui, O. Lakhal, B. Daachi, J. B. Mbede, and R. Merzouki, "Adaptive neural network control of a compact bionic handling arm," *IEEE/ASME Transactions on Mechatronics,* vol. 20, no. 6, pp. 2862-2875, 2015.

[59] A. Melingui, R. Merzouki, and J. Mbede, "Compact bionic handling arm control using neural networks," *Electronics letters,* vol. 50, no. 14, pp. 979-981, 2014.

[60] C. Escande, T. Chettibi, R. Merzouki, V. Coelen, and P. M. Pathak, "Kinematic calibration of a multisection bionic manipulator," *IEEE/ASME transactions on mechatronics,* vol. 20, no. 2, pp. 663-674, 2014.

[61] A. Grzesiak, R. Becker, and A. Verl, "The bionic handling assistant: a success story of additive manufacturing," *Assembly Automation,* 2011.

[62] H. Tsukagoshi, A. Kitagawa, and M. Segawa, "Active hose: An artificial elephant's nose with maneuverability for rescue operation," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, 2001, vol. 3: IEEE, pp. 2454-2459.

[63] H. Tsukagoshi, A. Kitagawa, and Y. Kamata, "Wearable fluid power composed of transformed flat tube actuators," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, vol. 2: IEEE, pp. 1178-1183.

[64] E. Guglielmino, N. Tsagarakis, and D. G. Caldwell, "An octopus anatomy-inspired robotic arm," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems,* 2010: IEEE, pp. 3091-3096.

[65] H. Jiang, X. Liu, X. Chen, Z. Wang, Y. Jin, and X. Chen, "Design and simulation analysis of a soft manipulator based on honeycomb pneumatic networks," in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2016: IEEE, pp. 350-356.

[66] H. Jiang *et al.*, "A two-level approach for solving the inverse kinematics of an extensible soft arm considering viscoelastic behavior," in *2017 IEEE*

International Conference on Robotics and Automation (ICRA), 2017: IEEE, pp. 6127-6133.

[67] M. M. Coad *et al.*, "Vine robots: Design, teleoperation, and deployment for navigation and exploration," *IEEE Robotics & Automation Magazine*, vol. 27, no. 3, pp. 120-132, 2019.

[68] E. W. Hawkes, L. H. Blumenschein, J. D. Greer, and A. M. Okamura, "A soft robot that navigates its environment through growth," *Science Robotics*, vol. 2, no. 8, p. eaan3028, 2017.

[69] P. E. Dupont, J. Lock, B. Itkowitz, and E. Butler, "Design and control of concentric-tube robots," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 209-225, 2009.

[70] R. J. Webster, J. M. Romano, and N. J. Cowan, "Mechanics of precurved-tube continuum robots," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 67-78, 2008.

[71] T. K. Morimoto, E. W. Hawkes, and A. M. Okamura, "Design of a compact actuation and control system for flexible medical robots," *IEEE robotics and automation letters*, vol. 2, no. 3, pp. 1579-1585, 2017.

[72] A. D. Marchese, R. K. Katzschmann, and D. Rus, "A recipe for soft fluidic elastomer robots," *Soft robotics*, vol. 2, no. 1, pp. 7-25, 2015.

[73] R. K. Katzschmann, A. D. Marchese, and D. Rus, "Autonomous object manipulation using a soft planar grasping manipulator," *Soft robotics*, vol. 2, no. 4, pp. 155-164, 2015.

[74] A. D. Marchese, K. Komorowski, C. D. Onal, and D. Rus, "Design and control of a soft and continuously deformable 2d robotic manipulation system," in *2014 IEEE international conference on robotics and automation (ICRA)*, 2014: Ieee, pp. 2189-2196.

[75] L. Wang, D. Zheng, P. Harker, A. B. Patel, C. F. Guo, and X. Zhao, "Evolutionary design of magnetic soft continuum robots," *Proceedings of the National Academy of Sciences*, vol. 118, no. 21, p. e2021922118, 2021.

[76] C. Zhou *et al.*, "Ferromagnetic soft catheter robots for minimally invasive bioprinting," *Nature communications*, vol. 12, no. 1, pp. 1-12, 2021.

[77] Y. Kim, G. A. Parada, S. Liu, and X. Zhao, "Ferromagnetic soft continuum robots," *Science Robotics*, vol. 4, no. 33, p. eaax7329, 2019.

[78] L. Wang, Y. Kim, C. F. Guo, and X. Zhao, "Hard-magnetic elastica," *Journal of the Mechanics and Physics of Solids*, vol. 142, p. 104045, 2020.

[79] M. T. Chikhaoui, A. Benouhiba, P. Rougeot, K. Rabenorosoa, M. Ouisse, and N. Andreff, "Developments and control of biocompatible conducting polymer for intracorporeal continuum robots," *Annals of Biomedical Engineering*, vol. 46, no. 10, pp. 1511-1521, 2018.

[80] A. Benouhiba, K. Rabenorosoa, P. Rougeot, M. Ouisse, and N. Andreff, "A multisegment electro-active polymer based milli-continuum soft robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018: IEEE, pp. 7500-7506.

[81] W. McMahan, B. A. Jones, and I. D. Walker, "Design and implementation of a multi-section continuum robot: Air-octor," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005: IEEE, pp. 2578-2585.

[82] G. Immega and K. Antonelli, "The KSI tentacle manipulator," in *Proceedings of 1995 IEEE international conference on robotics and automation*, 1995, vol. 3: IEEE, pp. 3149-3154.

[83] G. Immega, K. Antonelli, and J. Ko, "Teleoperation of the KSI Tentacle Manipulator for hot cell decontamination," in *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, 1995, vol. 3: IEEE, pp. 2133-2136.

[84] T.-D. Nguyen and J. Burgner-Kahrs, "A tendon-driven continuum robot with extensible sections," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015: IEEE, pp. 2130-2135.

[85] E. Amanov, T.-D. Nguyen, and J. Burgner-Kahrs, "Tendon-driven continuum robots with extensible sections—A model-based evaluation of path-following motions," *The International Journal of Robotics Research*, vol. 40, no. 1, pp. 7-23, 2021.

[86] T. Zhang, L. Yang, X. Yang, R. Tan, H. Lu, and Y. Shen, "Millimeter‐Scale Soft Continuum Robots for Large‐Angle and High‐Precision Manipulation by Hybrid Actuation," *Advanced Intelligent Systems*, vol. 3, no. 2, p. 2000189, 2021.

[87] M. Abdel-Nasser and O. Salah, "New continuum surgical robot based on hybrid concentric tube-tendon driven mechanism," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 235, no. 24, pp. 7550-7568, 2021.

[88] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43-55, 2006.

[89] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, p. e00938, 2018.

[90] A. Melingui, R. Merzouki, J. B. Mbede, C. Escande, and N. Benoudjit, "Neural networks based approach for inverse kinematic modeling of a compact bionic handling assistant trunk," in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*, 2014: IEEE, pp. 1239-1244.

[91] A. Melingui, C. Escande, N. Benoudjit, R. Merzouki, and J. B. Mbede, "Qualitative approach for forward kinematic modeling of a compact bionic handling assistant trunk," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 9353-9358, 2014.

[92] M. Giorelli, F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi, "Neural network and jacobian method for solving the inverse statics of a cable-driven soft arm with nonconstant curvature," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 823-834, 2015.

[93] A. Kapadia and I. D. Walker, "Task-space control of extensible continuum manipulators," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011: IEEE, pp. 1087-1092.

[94] E. Tatlicioglu, I. D. Walker, and D. M. Dawson, "Dynamic modelling for planar extensible continuum robot manipulators," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007: IEEE, pp. 1357-1362.

[95] W. S. Rone and P. Ben-Tzvi, "Continuum robot dynamics utilizing the principle of virtual power," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 275-287, 2013.

[96] J. Till, V. Aloi, and C. Rucker, "Real-time dynamics of soft and continuum robots based on Cosserat rod models," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 723-746, 2019.

[97] A. A. Alqumsan, S. Khoo, and M. Norton, "Robust control of continuum robots using Cosserat rod theory," *Mechanism and Machine Theory*, vol. 131, pp. 48-61, 2019.

[98] K.-H. Lee *et al.*, "Nonparametric online learning control for soft continuum robot: An enabling technique for effective endoscopic navigation," *Soft robotics*, vol. 4, no. 4, pp. 324-337, 2017.

[99] S. Hirai, "Measuring McKibben actuator shrinkage using fiber sensor," in *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2015: IEEE, pp. 628-633.

[100] W. Felt and C. D. Remy, "Smart braid: Air muscles that measure force and displacement," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014: IEEE, pp. 2821-2826.

[101] J. Morrow *et al.*, "Improving soft pneumatic actuator fingers through integration of soft sensors, position and force control, and rigid fingernails," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016: IEEE, pp. 5024-5031.

[102] H. Wang, M. Totaro, and L. Beccai, "Toward perceptive soft robots: Progress and challenges," *Advanced Science*, vol. 5, no. 9, p. 1800541, 2018.

[103] T. Helps and J. Rossiter, "Proprioceptive flexible fluidic actuators using conductive working fluids," *Soft robotics*, vol. 5, no. 2, pp. 175-189, 2018.

[104] S. Russo, T. Ranzani, H. Liu, S. Nefti-Meziani, K. Althoefer, and A. Menciassi, "Soft and stretchable sensor using biocompatible electrodes and liquid for medical applications," *Soft robotics*, vol. 2, no. 4, pp. 146-154, 2015.

[105] C. Larson *et al.*, "Highly stretchable electroluminescent skin for optical signaling and tactile sensing," *science*, vol. 351, no. 6277, pp. 1071-1074, 2016.

[106] Y. Chen, J. M. Oliveira, and I. W. Hunter, "Two-axis bend sensor design, kinematics and control for a continuum robotic endoscope," in *2013 IEEE International Conference on Robotics and Automation*, 2013: IEEE, pp. 704-710.

[107] G. Gerboni, A. Diodato, G. Ciuti, M. Cianchetti, and A. Menciassi, "Feedback control of soft robot actuators via commercial flex bend sensors," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 4, pp. 1881-1888, 2017.

[108] W. S. Rone and P. Ben-Tzvi, "Multi-segment continuum robot shape estimation using passive cable displacement," in *2013 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, 2013: IEEE, pp. 37-42.

[109] S. Sareh, Y. Noh, M. Li, T. Ranzani, H. Liu, and K. Althoefer, "Macrobend optical sensing for pose measurement in soft robot arms," *Smart Materials and Structures*, vol. 24, no. 12, p. 125024, 2015.

[110] S. Sareh *et al.*, "Bio-inspired tactile sensor sleeve for surgical soft manipulators," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014: IEEE, pp. 1454-1459.

[111] M. K. Dobrzynski, R. Pericet-Camara, and D. Floreano, "Contactless deflection sensor for soft robots," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011: IEEE, pp. 1913-1918.

[112] S. C. Ryu and P. E. Dupont, "FBG-based shape sensing tubes for continuum robots," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014: IEEE, pp. 3531-3537.

[113] R. Xu, A. Yurkewich, and R. V. Patel, "Curvature, torsion, and force sensing in continuum robots using helically wrapped FBG sensors," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 1052-1059, 2016.

[114] N. Rahman, N. J. Deaton, J. Sheng, S. S. Cheng, and J. P. Desai, "Modular FBG bending sensor for continuum neurosurgical robot," *IEEE robotics and automation letters*, vol. 4, no. 2, pp. 1424-1430, 2019.

[115] Y. Chitalia, N. J. Deaton, S. Jeong, N. Rahman, and J. P. Desai, "Towards FBG-based shape sensing for micro-scale and meso-scale continuum robots with large deflection," *IEEE robotics and automation letters*, vol. 5, no. 2, pp. 1712-1719, 2020.

[116] N. D. Inc. "Electromagnetic Tracking that Supports In-vivo Tool Navigation." https://www.ndigital.com/products/3d-guidance/ (accessed.

[117] M. Hoffmann *et al.*, "Electrophysiology catheter detection and reconstruction from two views in fluoroscopic images," *IEEE transactions on medical imaging*, vol. 35, no. 2, pp. 567-579, 2015.

[118] A. Vandini, C. Bergeles, F.-Y. Lin, and G.-Z. Yang, "Vision-based intraoperative shape sensing of concentric tube robots," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015: IEEE, pp. 2603-2610.

[119] R. Sirohi, Y. Wang, S. Hollenberg, I. S. G. Keith, I. D. Walker, and E. Green, "Design and characterization of a novel, continuum-robot surface for the human environment," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019: IEEE, pp. 1169-1174.

[120] J. M. Croom, D. C. Rucker, J. M. Romano, and R. J. Webster, "Visual sensing of continuum robot shape using self-organizing maps," in *2010 IEEE International Conference on Robotics and Automation*, 2010: IEEE, pp. 4591-4596.

[121] B. Ouyang, Y. Liu, H.-Y. Tam, and D. Sun, "Design of an interactive control system for a multisection continuum robot," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 5, pp. 2379-2389, 2018.

[122] D. B. Camarillo, K. E. Loewke, C. R. Carlson, and J. K. Salisbury, "Vision based 3-D shape sensing of flexible manipulators," in *2008 IEEE International Conference on Robotics and Automation*, 2008: IEEE, pp. 2940-2947.

[123] H. Guo *et al.*, "Continuum robot shape estimation using permanent magnets and magnetic sensors," *Sensors and Actuators A: Physical*, vol. 285, pp. 519-530, 2019.

[124] D. Stoyanov, M. V. Scarzanella, P. Pratt, and G.-Z. Yang, "Real-time stereo reconstruction in robotically assisted minimally invasive surgery," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2010: Springer, pp. 275-282.

[125] G. Dwyer *et al.*, "A continuum robot and control interface for surgical assist in fetoscopic interventions," *IEEE robotics and automation letters*, vol. 2, no. 3, pp. 1656-1663, 2017.

[126] A. Yeshmukhametov, K. Koganezawa, Y. Yamamoto, Z. Buribayev, Z. Mukhtar, and Y. Amirgaliyev, "Development of Continuum Robot Arm and Gripper for Harvesting Cherry Tomatoes," *Applied Sciences*, vol. 12, no. 14, p. 6922, 2022.

[127] R. Reilink, S. Stramigioli, and S. Misra, "3D position estimation of flexible instruments: marker-less and marker-based methods," *International journal of computer assisted radiology and surgery*, vol. 8, no. 3, pp. 407-417, 2013.

[128] G. Fang *et al.*, "Vision-based online learning kinematic control for soft robots using local gaussian process regression," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1194-1201, 2019.

[129] J. D. Greer, T. K. Morimoto, A. M. Okamura, and E. W. Hawkes, "Series pneumatic artificial muscles (sPAMs) and application to a soft continuum robot," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017: IEEE, pp. 5503-5510.

[130] É. Marchand and F. Chaumette, "Virtual Visual Servoing: a framework for real‐time augmented reality," in *Computer Graphics Forum*, 2002, vol. 21, no. 3: Wiley Online Library, pp. 289-297.

[131] T. Poignonec, P. Zanne, B. Rosa, and F. Nageotte, "Towards in situ backlash estimation of continuum robots using an endoscopic camera," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4788-4795, 2020.

[132] S. I. A. I. Center and G. J. Agin, *Real time control of a robot with a mobile camera*. SRI International, 1979.

[133] G. Flandin, F. Chaumette, and E. Marchand, "Eye-in-hand/eye-to-hand cooperation for visual servoing," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, 2000, vol. 3: IEEE, pp. 2741-2746.

[134] A. Muis and K. Ohnishi, "Eye-to-hand approach on eye-in-hand configuration within real-time visual servoing," *IEEE/ASME transactions on Mechatronics*, vol. 10, no. 4, pp. 404-410, 2005.

[135] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE transactions on robotics and automation*, vol. 12, no. 5, pp. 651-670, 1996.

[136] M. Chen, Y. Tang, X. Zou, Z. Huang, H. Zhou, and S. Chen, "3D global mapping of large-scale unstructured orchard integrating eye-in-hand stereo vision and SLAM," *Computers and Electronics in Agriculture*, vol. 187, p. 106237, 2021.

[137] H. Niu, Z. Ji, Z. Zhu, H. Yin, and J. Carrasco, "3d vision-guided pick-and-place using kuka lbr iiwa robot," in *2021 IEEE/SICE International Symposium on System Integration (SII)*, 2021: IEEE, pp. 592-593.

[138] A. A. Hayat, R. A. Boby, and S. K. Saha, "A geometric approach for kinematic identification of an industrial robot using a monocular camera," *Robotics and computer-integrated manufacturing*, vol. 57, pp. 329-346, 2019.

[139] R. S. Penning, J. Jung, N. J. Ferrier, and M. R. Zinn, "An evaluation of closed-loop control options for continuum manipulators," in *2012 IEEE International Conference on Robotics and Automation*, 2012: IEEE, pp. 5392-5397.

[140] D. Jakes, Z. Ge, and L. Wu, "Model-less active compliance for continuum robots using recurrent neural networks," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019: IEEE, pp. 2167-2173.

[141] F. Gul, W. Rahiman, and S. S. Nazli Alhady, "A comprehensive study for robot navigation techniques," *Cogent Engineering*, vol. 6, no. 1, p. 1632046, 2019.

[142] C. Stachniss, *Robotic mapping and exploration*. Springer, 2009.

[143] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," *Artificial intelligence review*, vol. 43, pp. 55-81, 2015.

[144] H. Cho, E. K. Kim, E. Jang, and S. Kim, "Improved positioning method for magnetic encoder type AGV using extended Kalman filter and encoder compensation method," *International Journal of Control, Automation and Systems*, vol. 15, pp. 1844-1856, 2017.

[145] H. Li and A. V. Savkin, "An algorithm for safe navigation of mobile robots by a sensor network in dynamic cluttered industrial environments," *Robotics and Computer-Integrated Manufacturing*, vol. 54, pp. 65-82, 2018.

[146] S. Lee and J.-B. Song, "Mobile robot localization using infrared light reflecting landmarks," in *2007 International Conference on Control, Automation and Systems*, 2007: IEEE, pp. 674-677.

[147] I. Ohya, A. Kosaka, and A. Kak, "Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing," *IEEE transactions on robotics and automation*, vol. 14, no. 6, pp. 969-978, 1998.

[148] M. Filipenko and I. Afanasyev, "Comparison of various slam systems for mobile robot in an indoor environment," in *2018 International Conference on Intelligent Systems (IS)*, 2018: IEEE, pp. 400-407.

[149] A. Garulli, A. Giannitrapani, A. Rossi, and A. Vicino, "Mobile robot SLAM for line-based environment representation," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005: IEEE, pp. 2041-2046.

[150] K. Yousif, A. Bab-Hadiashar, and R. Hoseinnezhad, "An overview to visual odometry and visual SLAM: Applications to mobile robotics," *Intelligent Industrial Systems*, vol. 1, no. 4, pp. 289-311, 2015.

[151] K. H. Sedighi, K. Ashenayi, T. W. Manikas, R. L. Wainwright, and H.-M. Tai, "Autonomous local path planning for a mobile robot using a genetic algorithm," in *Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753)*, 2004, vol. 2: IEEE, pp. 1338-1345.

[152] M. Alajlan, A. Koubaa, I. Chaari, H. Bennaceur, and A. Ammar, "Global path planning for mobile robots in large-scale grid environments using genetic algorithms," in *2013 International Conference on Individual and Collective Behaviors in Robotics (ICBR)*, 2013: IEEE, pp. 1-8.

[153] L. Cantelli, G. Muscato, M. Nunnari, and D. Spina, "A joint-angle estimation method for industrial manipulators using inertial sensors," *IEEE/ASME Transactions On Mechatronics*, vol. 20, no. 5, pp. 2486-2495, 2015.

[154] B. Munoz-Barron, J. R. Rivera-Guillen, R. A. Osornio-Rios, and R. J. Romero-Troncoso, "Sensor fusion for joint kinematic estimation in serial robots using encoder, accelerometer and gyroscope," *Journal of Intelligent & Robotic Systems*, vol. 78, pp. 529-540, 2015.

[155] D. Whitney, C. Lozinski, and J. M. Rourke, "Industrial robot forward calibration method and results," 1986.

[156] M. Keshmiri and W.-F. Xie, "Image-based visual servoing using an optimized trajectory planning technique," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 1, pp. 359-370, 2016.

[157] T. Shu, S. Gharaaty, W. Xie, A. Joubair, and I. A. Bonev, "Dynamic path tracking of industrial robots with high accuracy using photogrammetry sensor," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 3, pp. 1159-1170, 2018.

[158] J. Li, A. Ito, H. Yaguchi, and Y. Maeda, "Simultaneous kinematic calibration, localization, and mapping (SKCLAM) for industrial robot manipulators," *Advanced Robotics*, vol. 33, no. 23, pp. 1225-1234, 2019.

[159] M. Stilman, "Global manipulation planning in robot joint space with task constraints," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 576-584, 2010.

[160] H. Zhang, Y. Wang, J. Zheng, and J. Yu, "Path planning of industrial robot based on improved RRT algorithm in complex environments," *IEEE Access*, vol. 6, pp. 53296-53306, 2018.

[161] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," 1998.

[162] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566-580, 1996.

[163] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," in *Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512)*, 2000, vol. 1: IEEE, pp. 256-263.

[164] P. G. Zavlangas and S. G. Tzafestas, "Industrial robot navigation and obstacle avoidance employing fuzzy logic," *Journal of Intelligent and robotic Systems*, vol. 27, pp. 85-97, 2000.

[165] G.-C. Vosniakos and A. Chronopoulos, "Industrial robot path planning in a constraint-based computer-aided design and kinematic analysis environment," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 223, no. 5, pp. 523-533, 2009.

[166] K. Baizid *et al.*, "IRoSim: Industrial Robotics Simulation Design Planning and Optimization platform based on CAD and knowledgeware technologies," *Robotics and Computer-Integrated Manufacturing*, vol. 42, pp. 121-134, 2016.

[167] A. Mohammad, M. Russo, Y. Fang, X. Dong, D. Axinte, and J. Kell, "An efficient follow-the-leader strategy for continuum robot navigation and coiling," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7493-7500, 2021.

[168] H. Choset and W. Henning, "A follow-the-leader approach to serpentine robot motion planning," *Journal of Aerospace Engineering*, vol. 12, no. 2, pp. 65-73, 1999.

[169] Y. Chen, J. Liang, and I. W. Hunter, "Modular continuum robotic endoscope design and path planning," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014: IEEE, pp. 5393-5400.

[170] D. Palmer, S. Cobos-Guzman, and D. Axinte, "Real-time method for tip following navigation of continuum snake arm robots," *Robotics and Autonomous Systems,* vol. 62, no. 10, pp. 1478-1485, 2014.

[171] T. Kato, I. Okumura, S.-E. Song, A. J. Golby, and N. Hata, "Tendon-driven continuum robot for endoscopic surgery: Preclinical development and validation of a tension propagation model," *IEEE/ASME Transactions on Mechatronics,* vol. 20, no. 5, pp. 2252-2263, 2014.

[172] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE Journal on Robotics and Automation,* vol. 3, no. 4, pp. 323-344, 1987.

[173] M. works. "Camera Calibration." https://uk.mathworks.com/help/vision/camera-calibration.html (accessed 23 November 2018).

[174] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[175] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.