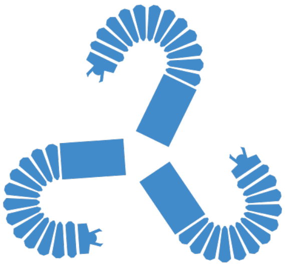
UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

DEPARTMENT OF COMPUTER SCIENCE

DEPARTMENT OF AUTOMATIC CONTROL

Surgical Subtask Automation for Intraluminal Procedures using Deep Reinforcement Learning



Author :
Ameya Pore

Supervisors :
Prof. Paolo Fiorini
Prof. Alicia Casals

Doctoral Program:
Computer Science, University of Verona, Cycle XXXV, S.S.D ING-INF/05
Department of Automatic Control, Universitat Politècnica de Catalunya, 2019

This dissertation is submitted in fullfillment of the requirements for the degree of
Doctor of Philosophy

June 2023

Surgical Subtask Automation for Intraluminal Procedures using Deep Reinforcement Learning
– Ameya Pore
PhD thesis
Verona, 18 March 2023

“One never notices what has been done; one can only see what remains to be done”
-Marie Sklodowska Curie

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 55 figures.

Author :	*Supervisor* :
Ameya Pore	**Prof. Paolo Fiorini**
	Prof. Alicia Casals
	June 2023

Acknowledgements

Like a ship that braves the raging seas,
This thesis hath faced triumphs and challenges with ease,
A journey long, perilous, and grand,
That would not have been possible without helping hands.

A multitude of individuals and organizations,
Whose contributions and support, in various situations,
Did make this journey less rough and tough,
Their insights and belief in my abilities, a great buff.

It is through their aid and unflinching care,
That this thesis hath become a reality, rare,
To them, I offer my gratitude and thanks,
For helping me reach the finish line, with their support and ranks.

I must start by thanking my wise guides,
Whose unwavering support did serve as my strides,
Prof. Paolo Fiorini of Verona's university,
And Prof. Alicia Casals from Catalonia's fraternity.

Their patience and guidance were of utmost worth,
Helping me navigate the complexities of this earth,
Enabling me to pursue this research journey with ease,
And complete this doctoral thesis with breeze.

Their knowledge and insights, oh so profound,
Were like a treasure trove that I always found,
And conversations with them, such a delight,
About entrepreneurship, history, and culture so bright.

My grateful heart extends its thanks,
To two wise mentors with scholarly ranks,
Dr. Diego Dall'Alba and Dr. Albert Hernansanz,
Whose guidance helped me navigate the research expanse.

With rigorous critique and constant vigilance they,
Molded the course of my work each day,
From research to administrative processes too,
They left no stone unturned to ensure all was true.

From greenhorn to budding scholar,
Dr Dall'Alba's tutelage was stellar.
His investment of time and expertise,
Instilled in me skills with ease.

Scientific writing, clarity in presentation,
And literature revision with no hesitation.
Critical and rational thinking became innate,
Experimental designing skills were now up to date.

I shall be forever grateful,
For his teachings that were so delightful.
The knowledge gained will always stay,
From greenhorn to budding scholar today.

My heart brims with utmost gratitude,
To Marie Skłodowska-Curie Actions, a shining fortitude,
Set by the European Commission, a beacon of light,
Who provided me an opportunity, a glorious sight.

In the ATLAS international training network, I found my place,
Their financial support, a heavenly grace,
Their aid in travel, and fellowship profound,
Helped me attend conferences, and schools abound.

Without their logistic and financial might,
This thesis, alas, would not have taken flight,
For their unwavering support, my thanks are due,
To Marie Skłodowska-Curie Actions, I give my gratitude anew.

Oh, let me sing my praises to the ATLAS consortium,
Whose watchful eye kept the project on its course,

Their coordinators, Dr. Emmanuel Vander Poorten and other PIs,
Did frequent reviews, keeping the team in force.

Though drafting deliverables, milestones, and reports,
Was no easy feat, with many a sleepless night,
We managed to submit them all before the deadline's court,
Guided by their feedback, ever shining bright.

ATLAS has weaved a tapestry of collaboration,
Among young researchers across the European nation,
Monthly meetings, events and journals were a routine,
Tying us together beyond colleagues, like a serene stream.

Zhen, Martina, Di, and Fernando, my trusted allies,
Together, our expertise made for interdisciplinary ties,
Their academic excellence and different know-hows,
Formed a perfect blend, like a painting's brushstrokes.

Amidst the ALTAIR LAB's bustling pace,
My gratitude to those in Office 1.64a,
Whose warmth and care filled every space,
Welcoming me into their Italiania.

From trattoria lunches to aperitivo nights,
They ensured my inclusion without hesitation,
Through language barriers, they showed me the sights,
And taught me to savor la dolce vita's elation.

Their efforts to make me a true Italian,
Are now reflected in my culinary finesse,
Ask me for recipes, I'll be your man,
And suggest a wine to perfect the caress.

To my dear colleagues, grazie mille,
Your kindness and friendship I shall always feel.

Fortunate I am to have friends so rare,
Sanat, Eleonora, Giovanni, who care
Despite our differences, cultural and far,
True friendship knows no boundaries, no bar.

Together we ventured, within Italy and beyond,
And our travels, so splendid, forever shall bond
Our motto, "Yet another trip" rings true and strong,
May our wanderlust never fade, may it forever prolong.

Barcelona's sojourn was a time I treasured,
For Maria and Mojtaba were with me to weather,
Their empathy and kindness created memories untold,
Together we laughed, joked, and made stories worth to behold.

Their presence made me forget the miles,
And we created a bond that still makes me smile,
I am grateful for the moments we shared,
And hope to see them soon, my dear friends who cared.

To the stalwarts of my life,
My parents and Dolo,
I owe an immense debt of gratitude,
For their unshakable belief in me,
Their love and support,
A constant source of strength and inspiration.

With their magical powers,
They can decipher my thoughts,
Without me ever having to utter a word,
Their empathy and understanding,
A beacon of hope, a soothing balm.

I am forever in their debt,
For their unconditional love and care,
My journey would be incomplete without them,
My hidden pillars, my rock, my solace.

In addition to the above, I must express
My immense gratitude for those who did bless
My life with guidance, motivation, and care
Who showed me the path and were always there

Mentors, teachers, colleagues, siblings, and friends
Who helped me surpass all the limits and bends
Who believed in me when I doubted myself
And encouraged me to strive for the best of myself

Their invaluable lessons and wise advice
Helped me achieve my dreams and rise
I owe them a debt that I can never repay
But I will cherish their love every single day.

Abstract

Intraluminal procedures have opened up a new sub-field of minimally invasive surgery that use flexible instruments to navigate through complex luminal structures of the body, resulting in reduced invasiveness and improved patient benefits. One of the major challenges in this field is the accurate and precise control of the instrument inside the human body. Robotics has emerged as a promising solution to this problem. However, to achieve successful robotic intraluminal interventions, the control of the instrument needs to be automated to a large extent.

The thesis first examines the state-of-the-art in intraluminal surgical robotics and identifies the key challenges in this field, which include the need for safe and effective tool manipulation, and the ability to adapt to unexpected changes in the luminal environment. To address these challenges, the thesis proposes several levels of autonomy that enable the robotic system to perform individual subtasks autonomously, while still allowing the surgeon to retain overall control of the procedure. The approach facilitates the development of specialized algorithms such as Deep Reinforcement Learning (DRL) for subtasks like navigation and tissue manipulation to produce robust surgical gestures. Additionally, the thesis proposes a safety framework that provides formal guarantees to prevent risky actions.

The presented approaches are evaluated through a series of experiments using simulation and robotic platforms. The experiments demonstrate that subtask automation can improve the accuracy and efficiency of tool positioning and tissue manipulation, while also reducing the cognitive load on the surgeon. The results of this research have the potential to improve the reliability and safety of intraluminal surgical interventions, ultimately leading to better outcomes for patients and surgeons.

Sommario

Le procedure intraluminali hanno aperto un nuovo sotto-campo della chirurgia minimamente invasiva che utilizza strumenti flessibili per navigare attraverso strutture luminali complesse del corpo, con conseguente riduzione dell'invasività e miglioramento dei benefici per i pazienti. Una delle principali sfide in questo campo è il controllo accurato e preciso dei dispositivi medici all'interno del corpo umano. La robotica è emersa come una soluzione promettente a questo problema. Tuttavia, per ottenere interventi intraluminali robotici di successo, il controllo del dispositivo medico deve essere automatizzato in larga misura.

La tesi esamina prima lo stato dell'arte nella robotica chirurgica intraluminale e identifica le sfide chiave in questo campo, che includono la necessità di una manipolazione degli strumenti sicura ed efficace e la capacità di adattarsi a cambiamenti imprevisti nell'ambiente luminali. Per affrontare queste sfide, la tesi propone diversi livelli di autonomia che consentono al sistema robotico di eseguire singole sottoattività autonomamente, consentendo comunque al chirurgo di mantenere il controllo generale della procedura. L'approccio consente di sviluppare algoritmi specializzati come Deep Reinforcement Learning (DRL) per sottoattività come la navigazione e la manipolazione dei tessuti per produrre gesti chirurgici robusti. Inoltre, la tesi propone un quadro di sicurezza che fornisce garanzie formali per prevenire azioni rischiose.

Gli approcci presentati vengono valutati attraverso una serie di esperimenti che utilizzano piattaforme di simulazione e robotiche. Gli esperimenti dimostrano che l'automazione delle sottoattività può migliorare l'accuratezza e l'efficienza del posizionamento degli strumenti e della manipolazione dei tessuti, riducendo anche il carico cognitivo sul chirurgo. I risultati di questa ricerca hanno il potenziale per migliorare l'affidabilità e la sicurezza degli interventi chirurgici intraluminali, portando infine a migliori risultati per pazienti e chirurghi.

Resumen

Los procedimientos intraluminales han abierto un nuevo subcampo de la cirugía mínimamente invasiva que utiliza instrumentos flexibles para navegar a través de estructuras luminales complejas del cuerpo, lo que resulta en una reducción de la invasividad y en beneficios para los pacientes. Uno de los principales desafíos en este campo es el control preciso y exacto de los dispositivos médicos dentro del cuerpo humano. La robótica ha surgido como una solución prometedora para este problema. Sin embargo, para lograr intervenciones intraluminales robóticas exitosas, el control del dispositivo médico debe estar automatizado en gran medida.

La tesis examina el estado del arte en robótica quirúrgica intraluminal e identifica los principales desafíos en este campo, que incluyen la necesidad de una manipulación segura y efectiva de herramientas y la capacidad de adaptarse a cambios inesperados en el entorno luminal. Para abordar estos desafíos, la tesis propone varios niveles de autonomía que permiten al sistema robótico realizar sub tareas individualmente de manera autónoma, al tiempo que permite al cirujano retener el control general del procedimiento. El enfoque permite el desarrollo de algoritmos especializados como el aprendizaje profundo por refuerzo (DRL) para sub tareas como la navegación y la manipulación de tejidos para producir gestos quirúrgicos robustos. Además, la tesis propone un marco de seguridad que proporciona garantías formales para prevenir acciones riesgosas.

Los enfoques presentados se evalúan mediante una serie de experimentos utilizando plataformas de simulación y robótica. Los experimentos demuestran que la automatización de sub tareas puede mejorar la precisión y eficiencia de la posición de la herramienta y la manipulación de tejidos, al tiempo que reduce la carga cognitiva en el cirujano. Los resultados de esta investigación tienen el potencial de mejorar la confiabilidad y seguridad de las intervenciones quirúrgicas intraluminales, lo que finalmente conduce a mejores resultados para pacientes y cirujanos.

Resum

Les intervencions intraluminals han obert un nou subcamp de la cirurgia mínimament invasiva que utilitza instruments flexibles per navegar a través de conductes complexes del cos, el que fa el procés menys invasiu i aporta beneficis per als pacients. Un dels principals reptes en aquest camp és el control precís dels dispositius mèdics dins del cos humà. La robòtica ha sorgit com una solució prometedora per a aquest problema. Però per aconseguir intervencions intraluminals robòtiques exitoses, el control del dispositiu mèdic ha d'estar automatitzat en gran mesura.

En aquesta tesi s'examina l'estat de l'art en robòtica quirúrgica intraluminal i s'identifiquen els principals reptes en aquest camp, que inclouen la necessitat d'una manipulació segura i efectiva de les eines i la capacitat d'adaptar-se a canvis inesperats en aquest entorn. Per abordar aquests reptes, la tesi proposa diversos nivells d'autonomia que permeten al sistema robòtic realitzar sub-tasques individualment de manera autònoma, i a la vegada permet al cirurgià mantenir el control del procediment. L'enfocament del treball permet el desenvolupament d'algorismes especialitzats com l'aprenentatge profund per reforç (DRL) per a sub-tasques com la navegació i la manipulació de teixits per produir gestos quirúrgics robustos. Addicionalment, la tesi proposa un marc de seguretat que ofereix garanties formals per prevenir accions arriscades.

Les aproximacions presentades són avaluades mitjançant una sèrie d'experiments utilitzant plataformes de simulació i robòtiques. Els experiments demostren que la automatització de sub-tasques pot millorar la precisió i l'eficiència de la posició de l'eina i la manipulació de teixits, reduint a la vegada la càrrega cognitiva del cirurgià. Els resultats d'aquesta la investigació tenen el potencial de millorar la confiabilitat i la seguretat de les intervencions quirúrgiques intraluminals, el que finalment dona lloc a millors resultats per a pacients i cirurgians.

Table of contents

List of figures **xvii**

List of tables **xxv**

1 Introduction **1**

1.1 Intraluminal procedures (IP) 3

1.1.1 Challenges 3

1.1.2 Snake-like Flexible Robots 4

1.1.3 Autonomous Intraluminal Surgery 4

1.1.4 Motion planning 6

1.2 Intraluminal Procedures (IP) subtask decomposition 7

1.3 Contributions 8

1.4 Thesis Structure 10

1.5 Conclusions 11

2 Robotic Autonomy for Intraluminal Procedure **13**

2.1 Ethical and regulatory considerations 13

2.2 Definition of surgical autonomy 14

2.3 Levels of autonomy for IP navigation 16

2.4 Robotic advancements 18

2.4.1 Endovascular interventions 19

2.4.2 Transanal IP 20

2.4.3 Transurethral and transvaginal IP 22

2.4.4 Transoral IP 23

2.4.5 Transnasal IP 24

2.5 Conclusions 24

3 Robotic reinforcement learning for Surgery: Background **26**

3.1 Reinforcement learning formulation 27

3.1.1 Markov Decision Process 27

3.1.2 Value Functions 28

3.1.3 Policy search 30
3.2 Challenges of robotic learning in IP 33
3.3 Tractability Through Representation and Prior Knowledge 35
3.3.1 Smart state-action discretization 35
3.3.2 Learning from Demonstration (LfD) 35
3.3.3 Prior Knowledge Through Task Structuring 37
3.4 Conclusions 38

I Endoscopic Navigation 40

4 Motion planning for Intraluminal Procedures 41
4.1 Literature Survey Methodology 42
4.2 Taxonomy on motion planning for IP navigation 43
4.2.1 Node-based algorithms 45
4.2.2 Sampling-based algorithms 51
4.2.3 Optimization-based algorithm 52
4.2.4 Learning-based algorithms 54
4.3 Evaluation environments 57
4.3.1 Simulation environment 58
4.3.2 Real Robotic setup 61
4.4 Limitations of Motion Planning (MP) and improvements 62
4.5 Conclusions 65

5 Colonoscopy Navigation using deep visuomotor control 67
5.1 Introduction 67
5.2 Automated control for colonoscopy 69
5.3 Simulation platform 70
5.3.1 Colon simulation 71
5.4 Deep Visuomotor control 73
5.5 Experimental Validation 76
5.5.1 Endoscopist data acquisition 76
5.5.2 Training DVC 77
5.5.3 Supervision 77
5.6 Results and discussion 78
5.7 Conclusion 82

6 Constrained Reinforcement Learning for Safe Colon Navigation 84
6.1 Introduction 84
6.2 Problem Statement 87

6.2.1 Overview of the safety Framework 87
6.3 Constrained Reinforcement Learning 87
6.3.1 Deep Reinforcement Learning 87
6.3.2 Constrained DRL and Lagrangian-PPO 90
6.4 Formal Verification 91
6.5 Experimental Validation 92
6.5.1 Experimental setup 92
6.5.2 Training results 93
6.5.3 Formal verification results 94
6.6 Conclusions 95

II Colorectal Cancer (CRC) detection 97

7 CRC diagnosis using Optical Coherence Tomography (OCT) scanning 98
7.1 Introduction 98
7.2 Related Works 100
7.3 Optimization based control formulation 102
7.3.1 System Specification 102
7.3.2 Hierarchical Quadratic formulation 103
7.4 Endoscope Control 104
7.4.1 Image position control 104
7.5 OCT probe control 105
7.5.1 Polyp detection 105
7.5.2 Probe specification and model 105
7.5.3 Polyp approach by OCT probe Image-based Visual Servoing (IBVS) . 105
7.5.4 OCT contact control 106
7.5.5 OCT tissue health classification 106
7.6 Experiments 107
7.6.1 Overview of the implementation 107
7.6.2 Results 108
7.7 Conclusion and Future work 110

III Tissue manipulation 112

8 Learning soft tissue manipulation 113
8.1 Introduction 113
8.2 *UnityFlexML*: a framework to learn surgical tasks in simulation 116
8.2.1 Robot Platform 116

8.2.2 Simulation Environment 117
8.3 Learning tissue retraction within *UnityFlexML* 117
8.3.1 Experimental Setup 118
8.3.2 Learning methods 120
8.3.3 Deep Reinforcement Learning (DRL) setting 120
8.3.4 Generative Adversarial Imitation Learning (GAIL) setting 121
8.3.5 Evaluation metrics 123
8.4 Autonomous tissue retraction in simulation 124
8.4.1 Results and Discussion 124
8.5 Sim-to-real autonomous tissue retraction 126
8.5.1 Results and Discussion 127
8.6 Conclusion 128

9 Safety in tissue manipulation using formal verification 130
9.1 Introduction 130
9.2 Safe-DRL for Tissue Retraction (TR) 132
9.2.1 Observation and Action Space 132
9.2.2 Reward Shaping 133
9.2.3 Training Algorithm 134
9.2.4 Formal Analysis 134
9.2.5 Violation Rate 135
9.3 Safety Evaluation 136
9.3.1 Results and Discussion 137
9.4 Conclusions 141

10 Conclusions and Future work 143
10.1 Future research directions 145
10.1.1 Robotic capabilities 145
10.1.2 Outstanding challenges in DRL and strategies for its mitigation 147

References 150

Appendix A Appendix: Hierarchical Task Decomposition using DRL for pick and place task 179

List of figures

1.1 IP with their clinical target. (a) Endovascular catheterization (b) Transanal colorectal procedures with a standard endoscope (c) Transurethral and transvaginal access for prostate or bladder procedures (d) Transoral procedures for airways or esophagus (e) Transnasal procedure to access bronchi. 2

1.2 Preoperative and intraoperative information is used to devise an interventional plan comprising a sequence of tasks and then execute it autonomously, replanning if necessary. A clinician always supervises the procedure and can take control at any time. Adapted from [1] 5

2.1 Case study of Levels of Autonomy (LoA) for endoscopic navigation for transanal IP. The complete navigation task is divided into three cognitive functions: target localization through preoperative imaging, planning the motion preoperatively and executing the motion. (Row 1) Target localization using preoperative images: The identified target is depicted with a red circle. (Row 2) Preoperative MP: Path representation inside the colon shown with yellow line. (Row 3) Motion execution intraoperatively: Intraoperative endoscopic visualization. (left to right). LoA 0-LoA 4 respectively. For each level, we indicate the agent that operates each cognitive function. Agent refers to either human operator, path-planning system or robotic manipulator. In case of two agents, the supervisor agent is depicted on the right side, while the main agent executing the actions is on the left. 19

2.2 Selection of some commercial robotic systems for IP. For endovascular interventions: Corpath™ system (Corindus, Waltham, USA) and Niobe™ system (Stereotaxis, St. Louis, USA), Sensei–Magellan (Hansen Medical, Mountain View, USA) and Monarch system (Auris Health, Redwood, USA); For transurethral and transvaginal procedures: Roboflex™ (ELMED, Ankara, Turkey) and Sensei–Magellan (Hansen Medical, Mountain View, USA); For gastrointestinal transanal procedures: Invendoscope™ (Invendo Medical, Weinheim, Germany) and Aer-O-Scope (GI View Ltd, Ramat Gan, Israel); For transnasal procedures: da Vinci (Intuitive Surgical, Sunnyvale, USA) and Flex® (Medrobotics, Raynham, USA) For bronchoscopic transoral intervention: Monarch system (Auris Health, Redwood, USA), ION™ (Intuitive Surgical, Sunnyvale, USA), da Vinci (Intuitive Surgical, Sunnyvale, USA) and Flex® (Medrobotics, Raynham, USA) are used. 21

3.1 Schematic illustration of a Convolutional Neural Network (CNN) based Deep Q-Network (DQN). The input to the network consists of a 128×128 image, followed by three convolutional layers, and a fully connected layer. The output of the fully connected layer is a probability distribution that indicates the action to take and the corresponding value of the state. 30

4.1 (a) Search matrix used for the survey (b) PRISMA flow diagram summarizing how the systematic review was conducted. 43

4.2 Schematics of the analysis carried out for each paper. These criteria include the targeted procedure, the level of autonomy, the motion planning method, the validation and the dynamics of the environment. 44

4.3 Classification of IP motion planning methods for continuum robots found in literature 45

4.4 Chronological development of endoluminal navigation. (a) Motion-planning approaches (b) The targeted IP procedures. Until 2010, the majority of studies have implemented node-based and sampling algorithms for MP. While lately, with the exponential increase in computational resources, the field is transitioning towards learning-based methods. 46

4.5 Schematics of simulation scenes used for endoluminal procedures. (A) Different models used for soft-object simulation [2–5], (B) Different simulation anatomies used for catheter insertion procedures, with additional constraints such as Blood flow, catheter motion and reconstruction from patient-specific data [4, 6, 7](C) Colon and endoscope deformation model for simulated colonoscopy [8, 9]. Dynamic cardiovascular lumen: Image superposition [10] (D) Urethroscopy procedure simulation with the catheter in Simulation Open Framework Architecture (SOFA) [5] (E) 3D models for transnasal surgeries using MRI data and Computed Tomography (CT) scans [11] (F) Simplified Bronchoscopy environment designed in Gazebo [12]. 60

4.6 Experimental setups for testing motion control in intraluminal procedures (A) Proposed experimental setups with different aortic arches [13] (B) blood circulation with a pumper and phantom deformations with a string [14] (C) Bronchoscopy setup for testing the guidance system [15](D) Various phantoms for colonoscopy [16] 61

5.1 Deep Visuomotor Control (Deep Visuomotor Control (DVC)) flow diagram. The environment provides a state observation S_t. The DVC agent uses the state input to generate an action a_t that is applied to the environment. During the training phase, DVC learns a task-conditioned policy π_ϕ to perform autonomous colonoscopy navigation. In the evaluation phase, the clinicians can supervise and override DVC decisions through action $a_{t'}$. 68

5.2 3D colon model construction. The 3D model is extracted from real patient CT scan to generate volumetric meshes. Mucosa textures from real endoscopy images are added to the meshes [17]. 71

5.3 Representation of the local frame at the endoscope tip. The X-Y plane of the camera is parallel to the image frame, while the z-axis represents the direction of insertion. Tip bending is carried out on the X-Y plane while the roll is carried on the z-axis. DVC uses a low-resolution image as state input. The green region represents the detected lumen center. 73

5.4 Proposed adaptive threshold segmentation pipeline for lumen detection. Each RGB frame captured by the endoscopic camera is passed through the adaptive filter to detect the dark pixels a) original RGB frame b) Image mask for the detected lumen (in green) c) distance vector between the image center P_c and the centroid of the detected darkest regions P_L. 74

5.5 Colon models used in the experimental phase. (From left to right) ranked in increasing complexity order, C_0, C_1, C_2 and C_3 colon models. The model complexity is characterized by the centerline distance of the model from rectum to caecum, and the number of acute bending, i.e. >90 degree, which is estimated through visual inspection. 76

5.6 (a) Navigation experiments with endoscopists (b) Novice user supervision while navigating by autonomous control strategies. *supervision* is printed on the screen, indicating the switch to manual control. When the endoscope is oriented towards the lumen (green point), the user can give back the control to the autonomous agent. A low-resolution (128x128 pixels) image is displayed to facilitate interpretability of machine decisions, however users have the option to change to high resolution (1024x1024 pixel) display. 77

5.7 Learning curve of DVC trained on varying complexity of colon, using three colon models. Cumulative reward is normalized in the range $[-1,1]$. The shaded area spans the range of values obtained when training the agent starting from five different initialization seeds. 79

5.8 Navigation performance comparison plots between DVC and endoscopists. Several parameters are plotted a) Lumen distance, b) perforations, c) Normalized distance, d) Time of insertion. 80

5.9 Trajectory plot of DVC, complex and smoothest endoscopist performance for a) C_1 b) C_2 3) C_3 models respectively. 81

6.1 Safe-Reinforcement Learning (RL) framework proposed in this work. Agents are trained in a Constrained Markov Decision Process (MDP) setting with soft constraints. The trained policies are examined with the Formal Verification (FV) tool which identifies the safety violations. Policies without safety violations are selected for final deployment to ensure a completely safe behavior. 86

6.2 An illustration of a capsule endoscope positioned inside the lumen facing an upcoming turn, showing the point of greatest depth (darkest point) and the lumen center. Navigation towards the deepest point can lead to biased motion towards the inner wall of the turn, potentially resulting in camera occlusion and collision with the wall, as demonstrated in (a) and (b). The right-hand side square boxes display the endoscopic view, with (a) depicting the similar views of two endoscope tips, while (b) shows that endoscope 1 approaches the wall more closely than endoscope 2 while following the line of greatest depth. 88

6.3 (a) Endoscopic view with the allowed actions. (b) Discrete representation of the input space used for the agent. 89

6.4 Illustration of four safety properties designed, namely $\Theta_{\downarrow}, \Theta_{\uparrow}, \Theta_{\rightarrow}$, and $\Theta_{\leftarrow}$. When the scope is close to the upper, lower, left, or right lumen wall, the respective row/column squares in the input space have high illumination with values in $[0.8, 1]$, hence the agent should not move in that direction. 91

6.5 Average performance vs the number of episodes of Proximal Policy Optimization (PPO) and Lagrangian Proximal Policy Optimization (L-PPO) over ten seeds (a) Average reward and (b) cumulative cost. Solid blue and red dashed lines are the empirical mean, while shaded regions represent the standard deviation. Black dashed line is the cost threshold 93

6.6 (a)Adversarial example discovered with FV for the safety property $\Theta_{\uparrow}$. (b) A small perturbation in the square marked green, the agent shows safe behavior. (c) Hard constraint violation positions for one of the PPO policies marked with green crosses. 95

7.1 (a) STRAS robotic setup. (b) Motorized OCT probe. 99

7.2 Overview of experimental setup depicting the realistic colon model, the lumen, OCT probe and the endoscope body. 100

7.3 Global navigation with information from endoscopic camera 104

7.4 Scanning strategies for colon lumen with robotized endoscopic OCT 107

7.6 a) Image guided control. Sequence of frames obtained from the monocular camera while aligning towards the polyp (i) in normal lighting conditions, (ii) in varying lighting conditions, (iii) Experiment two: translating and aligning towards the detected polyp. b) Top: Deployment of the OCT probe. Bottom: Data generated while tissue scanning. c) Plot of evolution of E_x,E_y and d_{OCT} for the second experiment performed. 109

8.1 *UnityFlexML* framework. The simulated da Vinci Surgical System (dVSS) arms interact with deformable tissues, modeled using Position-Based Dynamics (PBD) method. Example scene (a) at rest, (b) during tissue manipulation. . . 116

8.2 (a) In our setup the Patient Side Manipulator (PSM) interacts with silicone fat tissue covering a kidney phantom. The simulated scene controlling PSM movements within *UnityFlexML* can be seen in the background. (b) The calibration board used to uniquely map all the components of our real experimental setup to the *UnityFlexML* environment. 118

8.3 One of the experiments of the optimization process. The fat tissue is anchored to the calibration board (right side in the figure). (a) Rest condition; (b) Deformed condition. Point cloud of the deformed tissue is acquired with the depth camera shown on the right. 119

8.4 Network architecture of GAIL and PPO. PPO consists of a policy (actor) and Value network. The policy network acts as Generator for GAIL. Generated trajectories and expert trajectories are passed to the Discriminator. Discriminator learns a probability function which classifies the generator trajectory as expert or non-expert. The network layer details are depicted inside each box in the format (hidden units, activation) respectively. 121

8.5 The proposed methodology of Learning from Demonstrations (LfD) for the tissue retraction surgical gesture. (a) Expert demonstrations are performed and recorded using the dVRK console (b) Robotic agent is trained within a simulated environment. (c) The learnt policy is translated to the real robotic system. 123

8.6 Sequence of action frames for task completion in simulation: (a) approach, (b) grasp, (c) retract, (d) expose. Perspective of the simulated camera is overlaid on the bottom left of each simulator frame. 124

8.7 The obtained learning curve for GAIL and PPO. Cumulative reward is normalized in the range $[-1,0]$. The shaded area spans the range of values obtained when training the agent starting from three different initialization seeds. 125

8.8 Simulation experiments: Tumor Exposure (TE) from the camera at different initial positions, of the PSM for (a) PPO, (b) GAIL. The color of each subregion is related to the percentage of visible tumor area when $\mathbf{p}_0$ belongs to that subregion. The fat boundary from the top view is depicted in red dashed lines whereas the fat attachment is shown in the solid red line 126

8.9 Sequence of action frames for task completion in real world setup, with the circular mask used to compute TE metrics. (a) approach (TE=0%), (b) grasp (TE=0%), (c) retract (TE=∼15%), (d) tumor exposure (TE=100%). The real camera is placed in front of the phantom, in the same position as in the simulation (which does not correspond to the viewpoint of these pictures). . . 126

8.10 Real grasp experiments: TE from the camera when starting from different initial positions of the End Effector (EE), using (a) PPO (b) GAIL. The portion of fat tissue which is not considered for the experiments is colored in gray. 127

9.1 Virtual scene used to simulate the TR task. (a) The yellow tissue represents the renal adipose tissue that needs to be retracted to expose the tumor (green sphere) embedded in the underlying kidney (not visible in the picture). (b) Explanatory overview of the safe EE workspace (light blue cylinder) and the mesh colliders (green lines) for the spinal column. 133

9.2 Explanatory output analysis of (left) decision-making problem with two outputs and one subdivision, and (right) output analysis with three outputs and multiple subdivisions 134

9.3 The obtained learning curves for Safe-PPO and Unsafe PPO. The curves are averaged over four different seeds and smoothed over 25 epochs. 138

9.4 Top-view of safe EE workspace showing the TE from different starting points (a) Unsafe-PPO (b) Safe-PPO. The marked circle shows the safe workspace projection, while the dashed pink line represents the attachment region. See text for more details. 138

9.5 (left) State values that cause a violation for Safe-PPO derived using the FV tool and (right) State distribution in a standard execution of Safe-PPO (1000 episodes). We describe the relationship between two-state inputs, i.e. normalized EE movement in the y-direction and target distance, to simplify visualization and use static values for other inputs. 141

A.1 Summary diagram of the hierarchical architecture proposed in this paper. The pick and place task is divided into Low-level Subtask Experts (Low-level Subtask Expert (LSE)), namely *approach*, *manipulate* and *retract*. These subtasks are coordinated using a High Level Choreographer (High Level Choreographer (HLC)). 181

A.2 Schematic overview of the LSE training and evaluation process: All the LSE are trained independently (from left to right) approach, manipulate and retract respectively. The LSE policy π is updated offline by sampling from a replay buffer after 300 steps using Deep Deterministic policy gradient (DDPG)+Hindsight Experience Replay (HER). The policy is evaluated after each epoch by using hand-engineered solution for other subtasks by computing the success rate on 100 episodes. 181

A.3 Pick and place task (a) accomplished with end-to-end learning strategy with DDPG+HER and our LSE DDPG+HER. (b) failure with a thin cylindrical object for end-to-end strategy (c) success with a narrow cylindrical object for the agent trained with our LSE strategy. (d) failure with a small box object for the agent trained with end-to-end strategy. (e) success with a small box object for the agent trained with our LSE strategy. 183

A.4 Different environments used for experiments (a) *PandaPickAndPlace-v0* (b) Franka Emika Robot used for real robot demonstrations. 183

A.5 Performance comparison of our training strategy using DDPG+HER and BC. Each experiment is executed independently three times with different seeds. Success is quantified as the percentage of successful grasp as a function of training steps. 185

A.6 LSE specialization analysis using different training strategies. Samples representing activation patterns using (a) hand-engineered solutions (b) learned using our subtask approach (c) learned using an end-to-end strategy for ten episodes. 186

List of tables

2.1 Descriptive classification of levels of autonomy of IP. H: Performed by a human operator, M: Performed by a machine. H/M: Performed by a human, assisted by a machine, M/H: Performed by a machine, assisted by a human. M^1: Performed under human supervision. 17

4.1 Background of path-planning methods. 49
4.2 Summary of motion planning methods for IPEI navigation 55
4.3 Summary of publications on simulation environment for endoluminal navigation 59

5.1 Navigation parameters used for validation with their description 75
5.2 NASA Task Load Index for novice users. Lower score indicate good user experience 80
5.3 Comparison between DVC and Endoscopists 81

6.1 Average distance traveled results. 93
6.2 Results of model selection. `SAT` indicates property violation 94

9.1 Considered policies used in the ablation study. 137
9.2 Summary of violation rates for each property 139

A.1 Performance of methods for the same level of success rate 186

List of Acronyms

ACO	Ant Colony Optimization
A3C	Advantage Actor-Critic
AI	Artificial Intelligence
ATLAS	AuTonomous intraLuminAl Surgery
BC	Behavioral Cloning
BFS	Breadth First Search
CBS	Centerline-based Structure
CCM	Constant Curvature Model
CMDP	Constrained Markov Decision Process
CNN	Convolutional Neural Network
CRC	Colorectal Cancer
CRL	Constrained Reinforcement Learning
CT	Computed Tomography
CTR	Concentric Tube Robot
DDPG	Deep Deterministic policy gradient
DFS	Depth First Search
DMPs	Dynamical Movement Primitives
DNN	Deep Neural Network
DoFs	Degrees-of-Freedom
DQN	Deep Q-Network
DPG	Deterministic Policy Gradient
DRL	Deep Reinforcement Learning
DVC	Deep Visuomotor Control
dVRK	da Vinci Research Kit

dVSS	da Vinci Surgical System
EE	End Effector
EM	ElectroMagnetic
EMR	Endoscopic Mucosa Resection
ESD	Endoscopic Submucosal Disection
ESR	Early-Stage Researcher
FE	Flexible Endoscope
FBG	Fiber Bragg Gratings
fURS	Robotic Flexible Ureteroscopy
FV	Formal Verification
GA	Genetic Algorithm
GAE	Generalized Advantage Estimation
GAN	Generative Adversarial Network
GAIL	Generative Adversarial Imitation Learning
GMM	Gaussian Mixture Model
GI	GastroIntestinal
HER	Hindsight Experience Replay
HMMs	Hidden Markov models
HLC	High Level Choreographer
HQP	Hierarchical Quadratic Programming
HRL	Hierarchical Reinforcement Learning
HTM	Homogeneous Transformation Matrix
IBVS	Image-based Visual Servoing
IEC	International Electrotechnical Commission
IP	Intraluminal Procedures
IRL	Inverse Reinforcement Learning
KL	Kullback–Leibler
LoA	Levels of Autonomy
LD	Lumen Distance
LfD	Learning from Demonstrations
LPA*	Lifelong Planning A*

L-PPO	Lagrangian Proximal Policy Optimization
L-BFGS	Limited-memory Broyden-Fletcher-Goldfarb-Shanno
LSE	Low-level Subtask Expert
LSTM	Long Short Term Memory
MDP	Markov Decision Process
MIS	Minimally Invasive Surgery
MP	Motion Planning
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error
QP	Quadratic Programming
OCT	Optical Coherence Tomography
PBD	Position-Based Dynamics
PPO	Proximal Policy Optimization
PRM*	Probabilistic RoadMap*
PSM	Patient Side Manipulator
RL	Reinforcement Learning
ROI	Region of Interest
ROS	Robot Operating System
RRG	Rapidly-exploring Random Graph
RRM	Rapidly-exploring RoadMap
RRT	Rapidly-exploring Random Tree
SAC	Soft Actor Critic
SLAM	Simultaneous Localization And Mapping
SOFA	Simulation Open Framework Architecture
STRAS	Single-access Transluminal Robotic Assistant for Surgeons
TD	Temporal Difference
TE	Tumor Exposure
TNE	Transnasal Endoscopy
TOE	Transoral Endoscopy
TOI	Time of Insertion
TRPO	Trust region policy optimization
TR	Tissue Retraction

Chapter 1

Introduction

The field of surgical intervention has been consistently evolving towards less invasive procedures, which have been made possible through the advancements in technology and engineering [18]. This pursuit has a rich history, dating back to the early innovations such as Bozzini's Cystoscope in 1805 [19], Desormeaux's endoscope in 1853, and Kelling's laparoscopy attempts in 1901 [20] [21]. The success of Minimally Invasive Surgery (MIS) has been a driving force behind the development of advanced technologies, particularly the rod endoscope developed by Hopkins in the 1960s, which was later improved with the integration of digital cameras [18].

MIS is a promising alternative to more invasive surgical interventions, such as open approach, as it significantly reduces the risks of associated mortality and morbidity. One of the main limitations of MIS is that the clinician's dexterity and sensory information are limited compared to open surgery [18]. This loss is because the instruments used in MIS are much smaller than those used in open surgery, and they are often inserted through narrow tubes, which limit the ability to manipulate the instruments with precision. Moreover, the tactile feedback provided by the instruments used in MIS is also limited, as the sense of touch is diminished due to the use of long, thin instruments. Therefore, robot-assisted MIS is a significant area of interest which utilizes rigid instruments with dexterous wrists and high-definition stereo vision systems. The distal dexterous wrists allow clinicians to carry out complex tissue manipulation tasks with greater ease than was previously possible with manual laparoscopic tools [22]. Despite the considerable progress in robot-assisted MIS, the adoption of rigid instruments limited their use to highly space-confined areas such as the gastrointestinal tract and lung pathways.

To overcome the limitations of rigid instruments in robot-assisted MIS, researchers, in recent years, have explored the use of snake-like devices [23]). IP are emerging medical therapies that use the natural lumens to access deep-seated regions of the body (Fig. 1.1). They are performed with the use of snake-like flexible instruments that can navigate through the complex intraluminal anatomy.

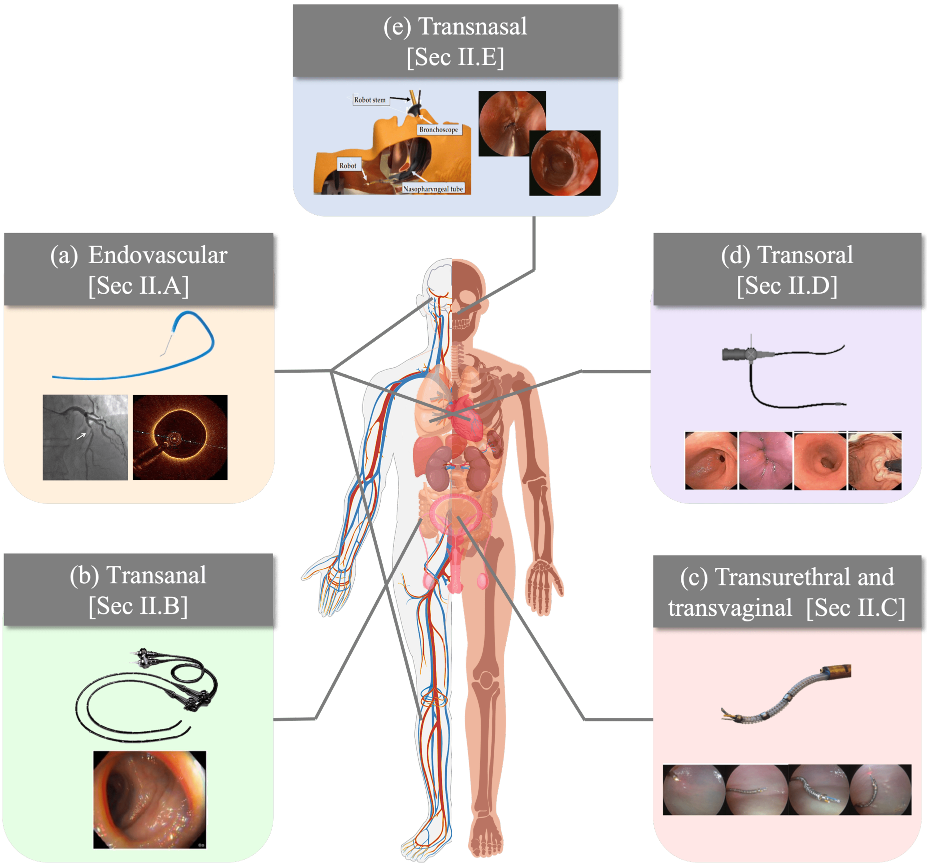

(e) Transnasal [Sec II.E]
Robot stem
Bronchoscope
Robot
Nasopharyngeal tube
(a) Endovascular [Sec II.A]
(d) Transoral [Sec II.D]
(b) Transanal [Sec II.B]
(c) Transurethral and transvaginal [Sec II.C]

Fig. 1.1 IP with their clinical target. (a) Endovascular catheterization (b) Transanal colorectal procedures with a standard endoscope (c) Transurethral and transvaginal access for prostate or bladder procedures (d) Transoral procedures for airways or esophagus (e) Transnasal procedure to access bronchi.

IP have demonstrated marked improvements in patient outcomes, including reduced blood loss, postoperative trauma, wound site infection, and hospitalization/recovery time [24]. However, the flexible tools used in IP have non-ergonomic designs and present difficulties in precise control due to the complex mapping between input and output motion. These design limitation leads to an increased cognitive and physical workload for the clinician [21]. Overall, it is a widely recognized fact that they need to undergo a long learning curve before becoming proficient in using such highly dexterous instruments [21].

To address these challenges, future generations of surgical robotics will likely provide more autonomous and dexterous control, resulting in enhanced accuracy, precision, and stability [25]).

1.1 Intraluminal procedures (IP)

IP can be classified into two categories: endoluminal and transluminal procedures [18, 22]. Endoluminal procedures are interventions in which instruments are inserted through and remain within natural body lumens or orifices. On the other hand, transluminal procedures involve the use of instruments within body lumens, but also incorporate incisions within lumen walls to access target sites beyond the lumen, as seen in Natural Orifice Transluminal Endoscopic Surgery.

Examples of endoluminal procedures include transoral interventions in the airways or esophagus, transanal access to the lower digestive tract, transnasal access to bronchi, and transurethral bladder and upper urinary tract procedures. Transluminal procedures include transgastric and transvaginal abdominal procedures, transoesophageal thoracic procedures, and transanal mesorectal procedures (Fig. 1.1).

In this thesis, the term IP is utilized in an inclusive manner to refer to both endoluminal and transluminal procedures.

1.1.1 Challenges

The implementation of IP often involves several complex and time-consuming phases, such as precise navigation to reach the targeted area, detection of abnormal tissue structures, and dissecting of the infected region [26, 27]. Such subtasks necessitate meticulous manipulation and control of the interventional instrument. One of the significant challenges during IP is related to operation in a deformable but confined workspace using a compliant device. The instruments used for IP must traverse the anatomical passageways while maintaining contact with the lumen along a significant portion of their length [24]. Such contact events occur beyond the field of view due to the restrictive perception of the endoluminal or endovascular tool architecture [22]. These contacts can be hazardous, and their response is usually challenging to predict, particularly as there is no direct view of the local anatomy. In addition, the movement of the tools is challenging to predict. Movement at the proximal end may lead to limited, unexpected or no movement of the distal tip [28]. The factors such as friction, slack, and deformation of the instrument and the vascular or luminal wall prevent a desirable 1-to-1 relation between the proximal and distal tip motion.

As a result, clinicians face a steep learning curve in terms of manipulating the tools within the body while observing a dislocated screen [24]. The visual feedback used in MIS is often two-dimensional and lacks depth perception, making it difficult to accurately assess the spatial relationships between different anatomical structures, creating situational awareness challenges [1]. To overcome these challenges, clinicians must rely on specialized training and image-guided navigation aids. Furthermore, the use of surgical tools instead of the hands results in a loss of sensory information regarding forces, texture, temperature, and stiffness [22, 29].

1.1.2 Snake-like Flexible Robots

The added constraints of IP place higher demands for robots that can provide distal dexterity in confined spaces. Concurrent developments enabling miniature camera technology have also been critical for advancing new miniature insertable visualization aids [30]. With the visualization challenges solved, the last decade has seen a flurry of research activity and new designs of snake-like robots for IP [22]. The mechanical architecture of flexible robots can be classified in three backbone types: continuous, discrete and hybrid [31]. Robots with continuous backbones (often referred to as continuum robots) use a continuous elastic backbone that is bent by wires, push-pull actuation or by antagonistic pairs of pre-shaped superelastic tubes. Robots with discrete backbones use articulated linkages, pivots and wire-compressed cams to form their structure. Hybrid backbone robots use a mixture of flexible elements (e.g. springs) and linkages to achieve manipulation.

Additionally, design constraints must be taken into account in IP. For instance, IP often require multiple robotic arms to operate through narrow access channels or anatomical passageways. This necessitates careful consideration of the design and mounting of the actuation units for each robotic arm to prevent collisions. Moreover, workspace constraints often dictate that the robotic arms emanate from a narrow access over-tube, imposing strict limitations on kinematic dexterity and workspace. A comprehensive survey of commercially available robotic systems for IP is presented in Chapter 2.

1.1.3 Autonomous Intraluminal Surgery

Despite the advancements in technology, the complexities associated with IP have not been fully mitigated by the introduction of robotic assistance. The limitations of current robotic systems, such as the lack of intuitive control due to poor shape-sensing capabilities and tool dexterity limitations, hinder their ability to effectively reduce the procedure's complexities [32, 1].

However, the potential benefits of automation in reducing clinicians' workload and improving the overall outcome of IP are widely acknowledged in the literature [32, 1, 33]. For example, the use of autonomous navigation assistance could help minimize path-related complications, such as perforation, embolisation, and dissection, caused by excessive interaction forces between the interventional tools and the lumen or vessels.

With the increasing demand for IP and the scarcity of specialists in this domain [29], the integration of autonomous control will enable clinicians to adopt a supervisory role and focus on high-level decisions instead of low-level execution. This approach can decrease the need for continuous human intervention, resulting in more efficient and streamlined procedures.

The use of automation in IP holds great potential for both patients and clinicians. As many IP actions are repetitive and fatiguing, IP can benefit from some degree of supervised autonomy where specific subtasks are delegated to the robot to perform autonomously under

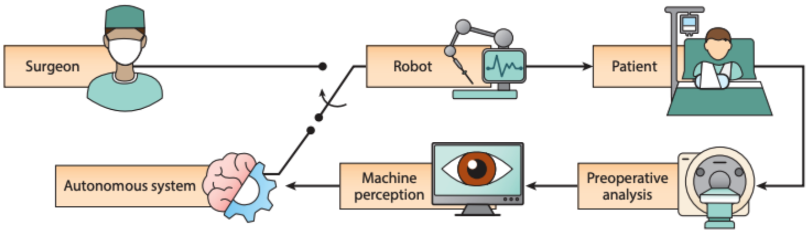


Fig. 1.2 Preoperative and intraoperative information is used to devise an interventional plan comprising a sequence of tasks and then execute it autonomously, replanning if necessary. A clinician always supervises the procedure and can take control at any time. Adapted from [1]

close supervision by a clinician. Such supervised autonomy can reduce the learning curve and increase accuracy while also reducing the risk of medical errors caused by human shortcomings, such as lack of attention, fatigue or poor decision-making [32].

Recently, a framework for the autonomy levels in robot-assisted MIS has been proposed, which includes different LoA such as robot assistance, task automation, conditional autonomy, and high-level autonomy [25]. Attanasio *et al.* and Haidegger *et al.* have carried out a comprehensive analysis of this framework and mapped out the distinct features of different LoA in robot-assisted MIS [33, 1].

Project details

This thesis lies within the AuTonomous intraLuminAl Surgery (ATLAS)[a] project that aims to provide autonomous control to smart, flexible robots to propel through complex deformable tubular structures. The project is a collaborative effort from seven prestigious university research groups, each of which serves as a center of excellence in a specific field, working together to seamlessly integrate various elements such as sensors, actuators, modeling, and control. Through the advancement of the state of the art in robotic surgery and relevant areas, including actuation and sensing for flexible instruments, automatic instrument localization, surgical workflow and operation state estimation, context-aware control schemes, and intuitive interface and guidance systems, the ATLAS project aims to make significant contributions to the field.

[a] The ATLAS project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 813782. The project establishes 15 early-stage researchers that are trained with the necessary skills and capabilities to understand complex surgical robotic challenges. More details of the project can be found at https://atlas-itn.eu/

However, these studies adopt a top-down approach to defining LoA based on general features of robot-assisted MIS, which makes it challenging to apply these levels to specific subtasks such as navigation in IP. To overcome this challenge, a bottom-up approach is required, which considers specific clinical phases and defines LoA based on characteristic features of subtasks associated with a specific clinical phase.

The inclusion of autonomous features raises several safety, ethical and regulatory concerns due to incorrect robot behavior. A framework that allows for safety validation in a wide variety of circumstances is necessary to overcome the regulatory requirements. Such a framework will enable tracking the source of error, detect deviations from expected functionality, and implement recovery action [34].

1.1.4 Motion planning

One of the initial steps towards enabling autonomous control for IP is to implement MP techniques [35]. MP refers to obtaining a path from a start to a goal configuration while respecting a collision-free workspace.

MP has been well-studied for rigid robotic manipulators [36] and recent studies have explored MP for flexible continuum robots with a large number of degrees of freedom [37, 38]. However, an organized survey of MP for IP and other biomedical applications using continuum robotic systems is missing. Therefore, in Chapter 4 of this thesis, we conduct a survey of existing MP methods for IP, the associated challenges and potential promising directions. The survey systematically classifies MP into four categories: node-based, sampling-based, optimization-based, and learning-based methods.

Based on the results of our survey, it is evident that there has been a marked rise in the adoption of learning-based methodologies that rely on neural network function approximators for acquiring motion representations. This surge can be attributed to the swift advancement in computational capabilities and parallel processing, which has made it feasible to estimate gradients over millions of parameters. In particular, data-driven learning techniques such as DRL hold significant promise as an alternative to the manual effort typically required for sequential decision-making processes [39]. DRL is an approach that identifies a sequence of actions aimed at increasing the likelihood of achieving a predefined objective.

DRL was first applied to Atari game environments [40, 41] but has recently been applied to robotics tasks such as locomotion skills, dexterous manipulation, and grasping [42–45]. In the healthcare domain, the DRL approach is being studied to produce optimum policies to suggest interventions and recommend actions, with the aim of reducing human-level bias and errors [46]. DRL can mimic a human-like learning approach and use electronic health records to develop treatment policies, intervention suggestion systems, and action recommendation systems [46]. The robustness of recent DRL algorithms helps developed systems adapt to sudden environmental changes.

Despite their increasing popularity, their application in the surgical robotics domain is limited due to safety, ethical, legal, and economic constraints. DRL has been applied for the manipulation of surgical needle [47], knot tying [48], cutting [49], and learning the tensioning policy of soft tissues [50]. However, the application of DRL in learning surgical tasks often present the policy with low-dimensional representation of observations such as robot kinematics data, which are widely accepted to be sample-efficient and trivial to learn [51–53]. This thesis aims to answer if it is possible to learn the policy directly from high-dimensional state inputs in an end-to-end manner.

In real-world scenarios, agents trained through DRL can encounter situations that are potentially dangerous or unsafe. Incorporating safety into DRL algorithms is, therefore, an essential research area to ensure that DRL agents operate safely and do not cause harm to themselves, other agents, or the environment. Incorporating safety into DRL involves designing algorithms that not only maximize rewards but also ensure that the agent behaves in a safe and responsible manner. This can involve constraining the agent's actions or placing limits on its exploration of the environment to avoid dangerous or undesirable outcomes. Therefore, one of the goals of this thesis is to develop a safety framework to ensure that the agent's action will be restricted to a pre-defined safety regime. The formal guarantee of a safe behavior is of utmost importance before their applicability in real surgical scenarios.

1.2 IP subtask decomposition

We choose transanal colonoscopy as a representative IP in this thesis, which involves the insertion of an endoscope, a flexible tube with a camera and light at the tip, into the lower digestive tract through the anus. The endoscope is used to visualize the interior lining of the rectum and colon to diagnose and treat a variety of conditions. Our choice for selecting transanal colonoscopy is motivated due to its ability to perform a variety of procedures such as biopsy, polyp removal and placement of stents and other devices, while being less invasive compared to other IP.

The workflow of transanal colonoscopy often includes three frequently used subtasks, namely navigation, abnormal tissue detection, and soft-tissue manipulation, which make up a significant portion of the overall procedure. Our goal in this thesis lies in automating these subtasks.

Navigation: The task of lumen navigation is a fundamental aspect of colonoscopy procedures and involves reaching the end of the lumen or the target area for inspection. The endoscopic camera provides visual feedback, which is utilized by the endoscopist to advance the instrument through the lumen. During routine screening procedures, a Flexible Endoscope (FE) is first inserted from the rectum to the caecum and then retracted to detect possible early-stage CRC lesions. A common gesture performed by endoscopists during the procedure is to centralize the endoscope towards the center of the lumen. Prior research has attempted to replicate this gesture through rule-based controllers that reduce the distance between the image center and the detected lumen center [54]. However, these algorithms are limited in situations where the endoscope tip approaches the colon wall, which can occur due to patient movements, peristalsis, or breathing. To address this issue, we propose an adaptive exploration method that leverages image-based DRL to determine the direction of motion.

Abnormal Tissue Detection: CRC lesions exhibit various features, such as texture, color, shape, borders, vessels, and size, which allow for classification into neoplastic and non-neoplastic polyps [55]. Polyps can be further categorized into diminutive (≤5 mm), small (6 to 9 mm), and large (≥10 mm) based on their size [55]. Although larger polyps are easier to visualize, they also pose a higher risk of malignancy [56]. Effective early detection of diminutive and small polyps is crucial to reduce the likelihood of their progression to large polyps. During a standard procedure, the endoscope tip is positioned in the vicinity of the target area, and a tissue biopsy is collected using forceps for histological analysis. An alternative to this method is offered by OCT which is a non-contact high-resolution imaging for polyp characterization [57]. In this thesis, we develop an OCT-based motion policy for tissue scanning that reduces the need for tissue removal while enabling real-time in-situ optical measurements to replace ex-situ biopsies.

Soft-tissue Manipulation: If polyp characterization results indicate a high probability of CRC, a polypectomy procedure is planned based on the polyp's location and features, such as Endoscopic Submucosal Disection (ESD) and Endoscopic Mucosa Resection (EMR). These procedures require the clinician to manipulate the mucosa tissue to identify the cutting plane, which requires grasping and retracting the tissue repeatedly. The choice of surgical system for dissection depends on the size and location of the polyp; for instance, the dVSS may be used for polyps closer to the anal opening, while flexible instruments may be used for polyps located farther away. In this thesis, we consider the dVSS as the system used for the polypectomy procedure due to its widespread usage. The tissue manipulation tasks carried out by the dVSS may require the clinician to switch robotic arms or instruct an assistant with the desired motion [58], making it an ideal candidate for automation using a DRL method. Hence, we present multiple approaches to automating the tissue manipulation task using DRL. It should be noted that while the automation discussed in this study is specifically implemented on the dVSS, which consists of rigid manipulator, the same principles and techniques can also be applied to flexible dual-arm robotic systems such as the Single-access Transluminal Robotic Assistant for Surgeons (STRAS).

1.3 Contributions

The purpose of this thesis is to explore the automation of repetitive surgical subtasks in IP to enhance the accuracy and safety of these procedures. This would result in a reduction in the physical burden on clinicians by delegating low-level actions to autonomous systems, thereby allowing human operators to focus on high-level decision-making tasks. The challenge lies in integrating partial automation of these subtasks into the surgical workflow in such a way that machines and humans can collaborate effectively in making decisions and executing actions.

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

1. A framework for classifying levels of dedicated autonomy for IP subtasks that provides predictable human intervention. The framework provides an intermediate LoA for the subtasks, which serves to clearly define the boundary between human and automated control, improving risk and safety management. Additionally, a comprehensive review of MP techniques for IP robot control is provided, which is an important step in achieving higher LoA. Limitations of current robotic systems and MP methods are also discussed, offering insight into areas for improvement.

2. Demonstration of end-to-end joint training for perception and control to learn colonoscopy navigation policies. The proposed method maps raw endoscopic images to the control signal of the endoscope, referred to as DVC. An open-source colonoscopy simulator incorporating deformable soft tissue dynamics was developed to support the proposed navigation method. The method was validated by comparison with data acquired from expert clinicians, and results showed equivalent navigation performance between DVC and expert clinicians. A novice user study was also conducted to demonstrate that supervision of DVC significantly reduces the user workload.

3. An autonomous robot control strategy that employs feedback from a monocular endoscopic camera and OCT imaging to detect malignant tissue and assess its health. The scanning strategy was demonstrated in a synthetic colon environment with varying lighting conditions and random image quadrants. The ability to perform real-time diagnosis of CRC using autonomous OCT scanning eliminates the need for ex-situ biopsy.

4. Application of DRL to automate soft tissue manipulation tasks. Additionally, a LfD training regime was explored, where expert demonstrations could be used to train robotic agents. DRL training was carried out in the open-source simulation framework *UnityFlexML*, and trained policies were transferred to the real robot.

5. A safe-DRL framework for incorporating safety constraints into the training process through constrained optimization. The safety of the robotic arms was evaluated using a FV and model selection tool, providing safety guarantees. The proposed method was used for colon navigation as well as soft tissue manipulation task.

6. Presentation of a multi-subtask RL methodology that allows complex tasks to be decomposed into low-level subtasks, which are then learned through DRL methods. A high-level choreographer combines the trained subtasks to achieve the intended task.

This thesis addresses limitations that currently prevent the widespread use of DRL methods in surgical subtask automation. The benefits of training DRL agents in a realistic simulation environment are emphasized, providing insight into the development of high-level autonomous IP systems. The contributions of this thesis have the potential to advance the development and performance of autonomous IP systems, bringing them closer to clinical application.

1.4 Thesis Structure

This thesis begins with a discussion of the necessity for robotic automation in IP in Chapter 2. The chapter starts with the significance of developing ethical and safety standards for effective risk management. The existing standards are presented and directions are provided to integrate autonomous features. Additionally, the autonomy framework developed for Robot-assisted MIS is also discussed and extended to include IP subtasks. Furthermore, the chapter presents a comprehensive overview of the commonly practiced IP, including transanal, transurethral, transoral, transnasal, and endovascular interventions. Moreover, it highlights the commercially available robotic systems used for these IP and compares their benefits to manual control and identifies their current limitations.

The core objective of this thesis is to demonstrate the use of RL for subtask automation in IP. Chapter 3 provides the mathematical background for framing the control problem in an RL context, and describes the challenges associated with robotics that make the application of RL a hard problem. We discuss the commonly used RL algorithms and LfD techniques for generating human-like surgical gestures.

The remainder of the thesis is divided into three parts, each focusing on a surgical subtask, elaborated in Sec. 1.2. One of the reasons to select the three frequently occuring subtasks was to show the versatility of the automation system across various task features. The subtasks have distinct objectives, showcasing the system's ability to generalize and adapt to diverse task requirements.

In Part 1, we concentrate on the IP navigation subtasks. Chapter 4 provides a literature survey of MP methods and summarizes the taxonomy of methods, results, limitations, and opportunities for improvement. In Chapter 5, we present an image-based endoscope navigation method called DVC, which maps endoscopic images to the endoscope's control signal. A realistic colonoscopy simulator with soft tissue dynamics is developed to perform end-to-end training. We use the environment to develop a safe-RL framework which adds constraints for safe colon navigation in Chapter 6. Several RL policies are trained and the one that adheres to all safety constraints in selected.

In Part 2, we consider the CRC detection subtask, where the aim is to develop detection techniques that would overcome the need for tissue removal. Chapter 7 shows the use of OCT imaging as a non-invasive screening technique to scan the suspected tissue and develops a control strategy to autonomously scan the tissue. The chapter presents a multi-objective optimization problem and demonstrates the use of quadratic programming in solving it. Future work would employ DRL methods developed in Part 1 to to solve the multi-objective setting.

Part 3 of the thesis focuses on the polypectomy procedure to excise the malignant CRC tissue. A subtask actively used in polypectomy procedures is tissue manipulation, which we aim to automate in this part. In Chapter 8, the tissue manipulation task is formulated in a RL problem where the agent learns the tissue dynamics from multiple interactions. The chapter employs a state-of-the-art DRL method to learn a tissue manipulation task in simulation, which can then be translated to a real robot. It further shows how simulation training can be performed using demonstrations collected from the real robotic system, replicating human-like gestures. Finally, Chapter 9 proposes a safe-RL method for manipulating soft tissue constrained within a pre-defined safety workspace. The safe behavior is validated using FV techniques.

The conclusions and future research directions are presented in Chapter 10.

One of the contribution of this thesis is the introduction of Hierarchical Reinforcement Learning (HRL) framework detailed in Appendix. 1, in which multiple agents can operate at different temporal levels to learn longer tasks. Lower-level agents learn to perform low-level actions to complete subtasks, while higher-level agents learn to sequence subtasks.

1.5 Conclusions

This chapter provides an introduction to IP and the challenges in instrument control. To address these challenges and reduce the physical burden on the operator, robotic systems under development aim to offer low-level motion control capabilities and improved dexterity while allowing the experts to retain supervisory control using intuitive interfaces.

It is essential to define the levels of autonomy where humans and machines can work together to make decisions and carry out actions. Existing autonomy frameworks for Robot-Assisted MIS have been proposed, but their application to IP is not straightforward. This thesis takes a bottom-up approach, where each subtask can have its own intermediate level of autonomy. Understanding the subtasks in this manner will enable better risk management by providing insight into the required level of human intervention.

Data-driven learning methods, such as DRL, are one way to automate surgical subtasks. In this thesis, DRL is used to demonstrate adaptable colonoscopy navigation and tissue manipulation skills that match the performance of human experts. The DRL agents are trained in a realistic simulation environment, and the trained policy is transferred to the real robot for validation using synthetic phantom experiments.

In addition, we develop a robot control strategy that uses an OCT sensor to scan and assess the health of malignant tissue in real-time. This reduces the need for tissue removal and could replace ex-situ biopsies with in-situ optical measurements.

Despite the potential of DRL in robotics, its application in surgery is limited by various safety, ethical, legal, and economic constraints. This thesis addresses some of these concerns by incorporating safety and human bias into the training process, making learning more efficient.

The results of this thesis represent early steps towards developing adaptive control for IP surgical systems. By sharing the simulation environment and methods developed in this thesis, we aim to encourage wider use of DRL in autonomous surgery.

Chapter 2

Robotic Autonomy for Intraluminal Procedure

One of the promising features of forthcoming IP robotic systems is autonomy, which grants the capability to automatically perceive, analyze, plan, and execute actions [59]. Autonomous robotic systems possess the capacity to handle non-programmed situations and exhibit self-management and self-guidance [60]. The most noteworthy feature of autonomy is the transfer of decision-making capacity from a human operator to the robotic system. Two conditions must be met for this transfer to occur [61]. Firstly, the operator must leave the control to the robotic system, including associated responsibilities (i.e., the human operator must demonstrate "trust" in the autonomous system). Secondly, the system must be validated, meaning that it must adhere to ethical, legal, and certification standards. However, these certification standards are not yet fully established for medical robotic systems due to a lack of consideration and a clear understanding of autonomy [62]. Therefore, this chapter first introduces the ethical and regulatory aspects of autonomy in Sec. 2.1, followed by a definition of generic Levels of Autonomy (LoA) in Sec. 2.2. We then present the specific LoA for IP navigation systems in Section 2.3. Finally, an overview of robotic advancements for various IP is provided in Sec. 2.4, followed by conclusions in Sec. 2.5.

2.1 Ethical and regulatory considerations

The ethical and regulatory considerations of autonomous medical robots can be addressed from multiple perspectives, including human rights, law, economics, policy, and ethics [63]. Medical robot practitioners have raised concerns about the potential consequences of errors resulting from decisions made by autonomous systems, which can be caused by incorrect robot behaviors [64]. Hence, the reliability of these robots is of utmost importance as any malfunction could lead to hazardous situations with risks of harming the patient or the medical personnel [64]. Reliability refers to the ability of the surgical robot to perform consistently and accurately over time, without compromising patient safety or compromising the effectiveness of the procedure.

The National Artificial Intelligence (AI) Initiative is one of the first research and development strategy to focus on promoting the responsible use of AI technologies in United States in various domains such as healthcare and medical robotics [65]. For medical robotics, this means developing systems that are accurate and reliable, while also ensuring that they are safe, transparent, accountable and used ethically and in a way that respects patient privacy. Similarly, the AI Act for high risk applications is a proposed regulation by the European Union that seeks to regulate the use of AI in certain high-risk applications [66]. The AI Act defines the role of a human operator, including the obligation to provide human supervision, the right for a human to override an automated decision, and the right to obtain human intervention, which forbids full autonomy. Therefore, human intervention needs to be carefully designed into the system at different levels of integration [67].

The reliability of medical robotics is associated with the notion of certification, which requires legal approval that the system has reached a particular standard. Several regulatory standards exist in the robotics domain, such as the International Electrotechnical Commission (IEC) Technical Report 60601-4-1 [68], which provides guidance towards risk management, basic safety, and essential performance towards systems with some degrees of autonomy. However, these standards are not fully developed for autonomous medical robotic systems due to a lack of clear understanding of autonomy and uncertainty [62, 64].

Fischer *et al.* identified key aspects to facilitate regulatory development, including architecture and engineering, requirements and specifications, and verification and validation issues [62]. Architecture and engineering issues entail making the autonomous system amenable to inspection and analysis, while requirements and specification issues define the expected behavior of the system and the intended goal. Verification and validation issues cover a broad range of techniques at different levels of formality. Upcoming regulations, such as the AI Act, are expected to drive earlier consideration of safety and system integration concerns in the design process. Machine decision interpretability, such as Explainable AI [69], can aid the forensic analysis of human-robot collaboration, and the use of formal methods, such as mathematical proofs of correctness, can improve validation and reliability.

The introduction of LoA could support regulatory development by providing different levels of system verification, validation, and improved risk management [62].

2.2 Definition of surgical autonomy

Quantifying the degree of autonomy in robotic systems can be challenging, as the capabilities of the robot can vary significantly based on the underlying technologies used. In the field of medical robotics, the introduction of autonomous capabilities in robotic systems is leading to a significant shift in the role of medical specialists. Traditionally, medical specialists have relied on their manual dexterity and interventional skills to diagnose and treat patients. However, as the capabilities of medical robotic systems continue to increase, medical specialists are increasingly shifting towards high-level decision-making tasks, relying on the robotic systems to provide assistance in the more repetitive and routine aspects of medical procedures.

Previous research has outlined five LoA for medical robotic systems, taking into account the complete clinical procedure and the role of the clinician [25, 33]. At LoA 0, the robot has no decision autonomy, and the human clinician controls all aspects of the system. At LoA 1, the robot can assist the clinician, while at LoA 2, it can independently perform an interventional subtask. At LoA 3, the robot can autonomously perform more extended segments of the clinical procedure while making low-level cognitive decisions. Finally, at LoA 4, the robotic system can execute the entire procedure based on human-approved clinical plans or surgical workflow. LoA 5, which refers to full autonomy where the robotic clinician can perform the entire procedure better than the human operator, is still in the realm of science fiction and outside the scope of this thesis [25, 33]. At highest LoA, the robotic system will exhibit highly sophisticated responses to a variety of sensory data, closely approaching a level of sensorimotor skills of an expert clinician.

The enabling technologies and the practical applications for different levels are outlined by Attanasio *et al.* in [1] while Haidegger *et al.* [33] provide a top-down classification of LoA for general robot-assisted MIS. Haidegger's classification considers four robot cognitive functions, including generate, execute, select, and monitor options, and the overall LoA is computed as the normed sum of the four system functions on a linear scale, ranging from 0 (fully manual) to 1 (fully autonomous).

In clinical practice, an interventional procedure workflow is typically decomposed into several granular levels, such as phases, steps, and gestures [70]. While many of the interventional phases and skills used in IP are not considered in robot-assisted MIS, such as luminal navigation, LoA defined for robot-assisted MIS cannot be directly applied to IP. Moreover, using the approach proposed by Haidegger *et al.*, it is challenging to identify a clear boundary between human and automated control that is required for specific phases/steps of robot-assisted MIS. This introduces an additional problem of defining the overall level of the system that implements different LoA for different phases of the procedure.

Therefore, we propose a bottom-up solution where an intermediate LoA is defined for specific interventional phases. Such an approach would provide a better comprehension of the level of human intervention necessary for specific sub-tasks. An assessment of individual stages can provide a more accurate overall estimation of the system's autonomy as subtasks can have varying intermediate LoAs. In this thesis, we consider one of the major subtask that occurs frequently in most IP i.e. navigation. Navigation is an elongated phase which consists of advancing the tip of the flexible instrument to reach the targeted area. Although, we apply the approach of intermediate LoA to a IP subtask, its application to other clinical subtasks, such as robot-assisted MIS, is straightforward.

2.3 Levels of autonomy for IP navigation

LoA for robot-assisted MIS has been derived from the degree of autonomy introduced by IEC in a technical report (IEC/TR 60601-4-1) [68] to propose an initial standardization of autonomy levels in medical robotics. The report parameterizes Degrees-of-Freedom (DoFs) along a system's four cognition-related functions: generate, execute, monitor and select options. A similar classification approach has been followed by Haidegger *et al.* for robot-assisted MIS. We identify three specific cognitive functions for an IP navigation task: 1) Target localization, 2) Motion planning MP, and 3) Execution and replanning. Target localization is usually based on preoperative images, such as CT, Magnetic Resonance Imaging (MRI) or X-Ray imaging. It is a critical feature, as inaccurate target identification can lead to errors in the subsequent steps. MP refers to the preoperative planning performed before the procedure. This may be done in static virtual models of the lumen or vessels. Execution and replanning is an intraoperative phase to carry out the required motion to reach the target while continuously replanning intraoperatively. It can include target relocalization when adjustment is needed due to unexpected situations.

Table 2.1 provides an overview of the LoA for IP. LoA 0 requires a human operator to take full control of target localization, MP and motion execution. Commercially available robotic system can be considered in this category since the human operator has complete control of the robotic motion. LoA 1 entails manual target localization and preoperative planning by a clinician with the robotic system providing assistance during motion execution. Examples of this level are systems that use external tracking devices and registration methods to align preoperative data with the intraoperative environment and aid the clinician with motion execution [71, 14, 72]. Taddese *et al.* developed a teleoperated magnetically controlled endoscope that provides navigation assistance through controlled magnetic fields [73]. The implementation of such systems represents LoA 1, where the manipulator executes the commands given by the operator.

In LoA 2, the robotic system fully controls the specific steps of navigation. Target localization is performed by the clinician, who provides input in the form of waypoints or demonstration trajectories. The path planner uses this information to generate a global trajectory, and the robotic system carries out the required motion indicated by the path planner. During execution, the human operator supervises the autonomous navigation and approves the robot's actions or overrides them. In LoA 3, the path planner generates the global path in the preoperative phase without any manual intervention after target localization by the clinician. LoA 3 also includes the ability to autonomously split the entire navigation task into specific subtasks. The robotic system executes the motion indicated by the path planner and adapts to environmental changes through real-time replanning. The local real-time knowledge provides information about the anatomical environment, and the motion is adjusted as the autonomously steering is performed. All features, including

Table 2.1 Descriptive classification of levels of autonomy of IP. H: Performed by a human operator, M: Performed by a machine. H/M: Performed by a human, assisted by a machine, M/H: Performed by a machine, assisted by a human. M^1: Performed under human supervision.

LoA	**Description**	**Target localization**	**Motion planning**	**Execution & re-planning**
0	*Direct robot control*: The clinician exclusively controls all cognitive functions without any support or assistance [25]. Most IP systems used in clinical practice operate at Level-0 autonomy.	H	H	H
1	*Navigation assistant*: The human operator maintains continuous control of the robotic navigation intraoperatively; however, it is assisted robotically during the execution of the motion. Other cognitive functions are carried out manually.	H	H or M	H/M
2	*Navigation using waypoints*: The operator provides discrete high-level navigation tasks such as waypoints or predefined trajectories. These trajectories are derived during preoperative planning. The robot carries out the required motion between the waypoints during the execution time, with the clinician in a supervisory role to approve or override the strategy.	H	M/H	M^1 or M/H
3	*Semi-autonomous navigation*: The final goal of navigation is provided by a human operator, and the system generates the strategies required to carry out the complete navigation task. During the execution time, it relies on the operator's supervision to approve or override the choice. In IP navigation, the robot would extract waypoints and then plan the trajectory to reach the point.	H	M	M^1
4	*High-level autonomous navigation*: This level is characterized by the ability of the system to make clinical decisions and execute the control solution under the clinician's supervision. The system should interpret preoperative imaging modalities such as CT, MRI and ultrasound to detect target regions and extract all the information required for proper navigation.	M	M	M^1

target localization, MP, and execution, are autonomously carried out without any human intervention by a system reaching LoA 4.

In LoA 4, the key addition is automatic target identification, which requires enabling technologies such as autonomous segmentation of organs, detection of abnormal tissues and automatic localization and shape sensing mechanisms [32]. Recent advancements in computer vision techniques have used deep learning models to segment tumor from CT scans [74]. This approach can be used to obtain a 3D model and location of the target, which can then be autonomously tracked in real-time. The robotic system can then execute the navigation plan during the procedure under the supervision of the human operator. An example of the proposed LoA for the transanal IP is presented in Fig. 2.1. In Chapter 4, the proposed LoA will be applied to classify all the works considered in the field of IP. Interested readers can refer to [75, 76] for a detailed overview of the computer vision techniques required for supporting the highest LoA.

2.4 Robotic advancements

A significant proportion of robotic systems developed for IP utilize continuum robots [22, 38]. Continuum robots are actuated structures that are composed of multiple segments that form curved shapes with continuous tangent vectors. They are considered to have an infinite number of joints and degrees of freedom, making them more flexible and adaptable to different anatomical configurations [38, 24]. The use of continuum robots in medical robotics has opened up new possibilities for performing interventions with improved accuracy and safety.

Despite their advantages, continuum robots are highly complex to model, sense, and control, posing significant challenges for their deployment in clinical practice [24]. To address these challenges, researchers are actively exploring new technologies to enhance the ability of these robots to recognize and interact with tissues with greater dexterity and sensory feedback [1]. These technological advances hold promise for improving the navigation guidance and building higher LoA.

Currently, some robotic systems are designed to be used in multiple procedures due to the lack of specific robotic technologies or the adaptability of the systems to different clinical scenarios [77]. In this section, we discuss the available robotic platforms for IP, focusing on common IP routes such as transnasal, transoral, transurethral, transanal procedures, and endovascular interventions, as target clinical applications. We exclude procedures in which the development of continuum robotic systems is in its infancy or where the navigation phase does not constitute the predominant phase, such as auditory canal access, transvascular interventions, and exploratory procedures of the lymphatic system. Fig. 2.2 provides an overview of the different IP routes considered in this section.

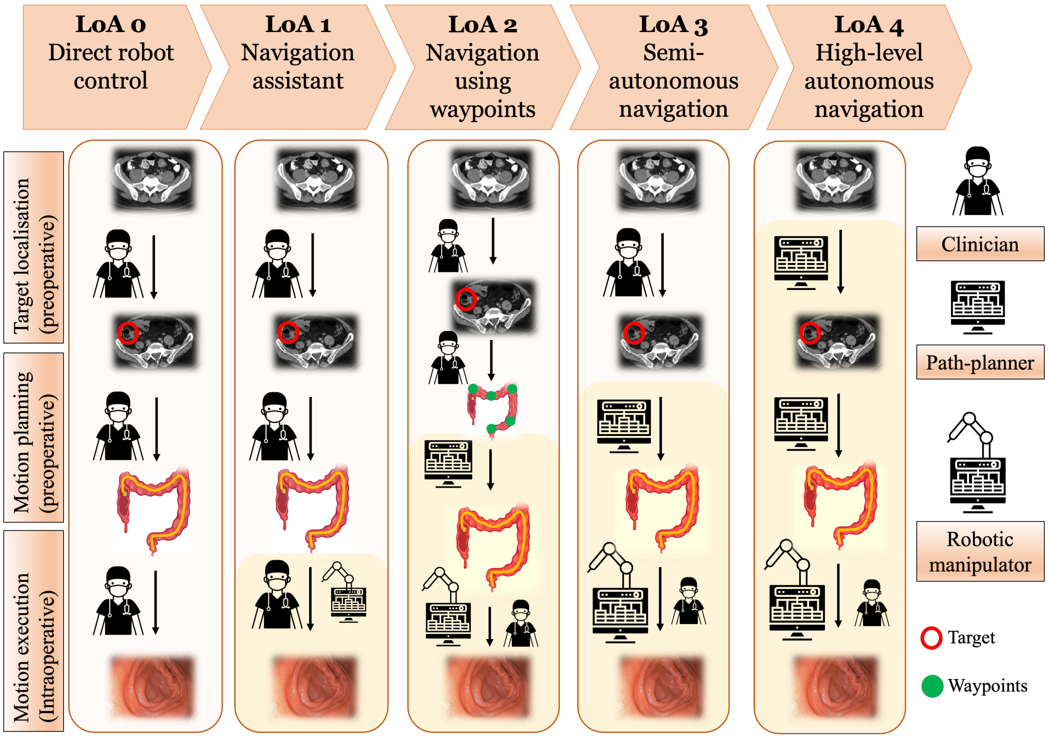

LoA 0
Direct robot control
LoA 1
Navigation assistant
LoA 2
Navigation using waypoints
LoA 3
Semi-autonomous navigation
LoA 4
High-level autonomous navigation
Target localisation (preoperative)
Motion planning (preoperative)
Motion execution (Intraoperative)
Clinician
Path-planner
Robotic manipulator
Target
Waypoints

Fig. 2.1 Case study of LoA for endoscopic navigation for transanal IP. The complete navigation task is divided into three cognitive functions: target localization through preoperative imaging, planning the motion preoperatively and executing the motion. (Row 1) Target localization using preoperative images: The identified target is depicted with a red circle. (Row 2) Preoperative MP: Path representation inside the colon shown with yellow line. (Row 3) Motion execution intraoperatively: Intraoperative endoscopic visualization. (left to right). LoA 0-LoA 4 respectively. For each level, we indicate the agent that operates each cognitive function. Agent refers to either human operator, path-planning system or robotic manipulator. In case of two agents, the supervisor agent is depicted on the right side, while the main agent executing the actions is on the left.

2.4.1 Endovascular interventions

Endovascular interventions are procedures in which a guidewire is introduced through a small incision in the groin, arm or neck, and is advanced to the desired location to act as a stable track for the catheter to follow. One of the major challenges in these procedures is the control of catheters and guidewires, particularly in terms of steering them through a 2D fluoroscopy image [78, 79]. Navigation is achieved through a combination of insertion, retraction, and application of torque actions at the proximal end of the catheter and guidewire. This can produce haptic feedback due to friction between the catheter and the vascular walls [80], requiring a precise understanding of the 3D anatomy projected in a 2D image plane.

Advancements in robotic technology, such as enhanced instrumentation, imaging, and navigation, have greatly improved the current state of endovascular procedures. Robotic platforms provide controlled steering of the catheter tip with improved stability, leading to a growing interest in teleoperated robotic catheterization systems. These systems offer reduced radiation exposure, increased precision, elimination of tremors, and improved operator comfort.

Recent developments in the CorPath™ GRX (Corindus, Waltham, USA) system provide guided robotic control that allows clinicians to navigate endovascular tools through a joystick. Other robotic catheter systems, such as the Sensei™ X and Magellan platforms, were introduced by Hansen Medical (Mountain View, USA) and later acquired by J&J robotics (New Brunswick, USA). Although they are not commercially available anymore, they are considered milestones in robotic systems for endovascular interventions [77]. Part of this technology was incorporated into the Monarch platform (Auris Health, Redwood, USA), which targets bronchoscopy.

The Hansen systems consist of a wire, a steerable inner leader, and a steerable outer guide, all of which attach to the robotic system at their proximal ends. Articulation is achieved using four pull wires (tendon-driven) through a 3D joystick or navigation buttons on the master workstation. The mechanically driven Amigo™ (Catheter Robotics Inc. Budd Lake, USA) and the R-One™ (Robocath, Rouen, France) robotic assistance platform allow standard catheters to be steered in 3 DoFs using an intuitive remote controller that replicates the standard handle of a catheter. The Niobe™ (Stereotaxis, St. Louis, USA) is a remote magnetic navigation system in which a magnetic field guides the catheter tip. The tip deflection is controlled by changing the orientation of outer magnets through a mouse or joystick at the master workstation.

Although these robotic systems have reported excellent intravascular navigation, the absence of haptic feedback affects the procedural outcome when maneuvering in smaller vessels such as coronary, cerebral, and visceral vessels [81, 82].

2.4.2 Transanal IP

Transanal colonoscopy is a commonly used procedure for the diagnosis and treatment of colonic diseases, including CRC [83, 28]. During a standard colonoscopy, a flexible tube is inserted through the anus and advanced to examine the colon wall [84]. Early detection and diagnosis of CRC lesions are crucial for improving patient outcomes [28, 84]. However, the increased workload of endoscopists has raised concerns about the ergonomic aspects of conventional colonoscopy. Studies have reported work-related musculoskeletal injuries among colonoscopists, including the hand, wrist, forearm, and shoulder [85, 86]. Although colonoscopy-related adverse events are rare, the proportion of subjects with risk factors is increasing. Severe colonoscopic complications such as perforation and bleeding can be fatal [87, 88]. Furthermore, even well-experienced endoscopists are often limited by the lack of maneuverability, which can result in around 20% of missed polyp localization [89].

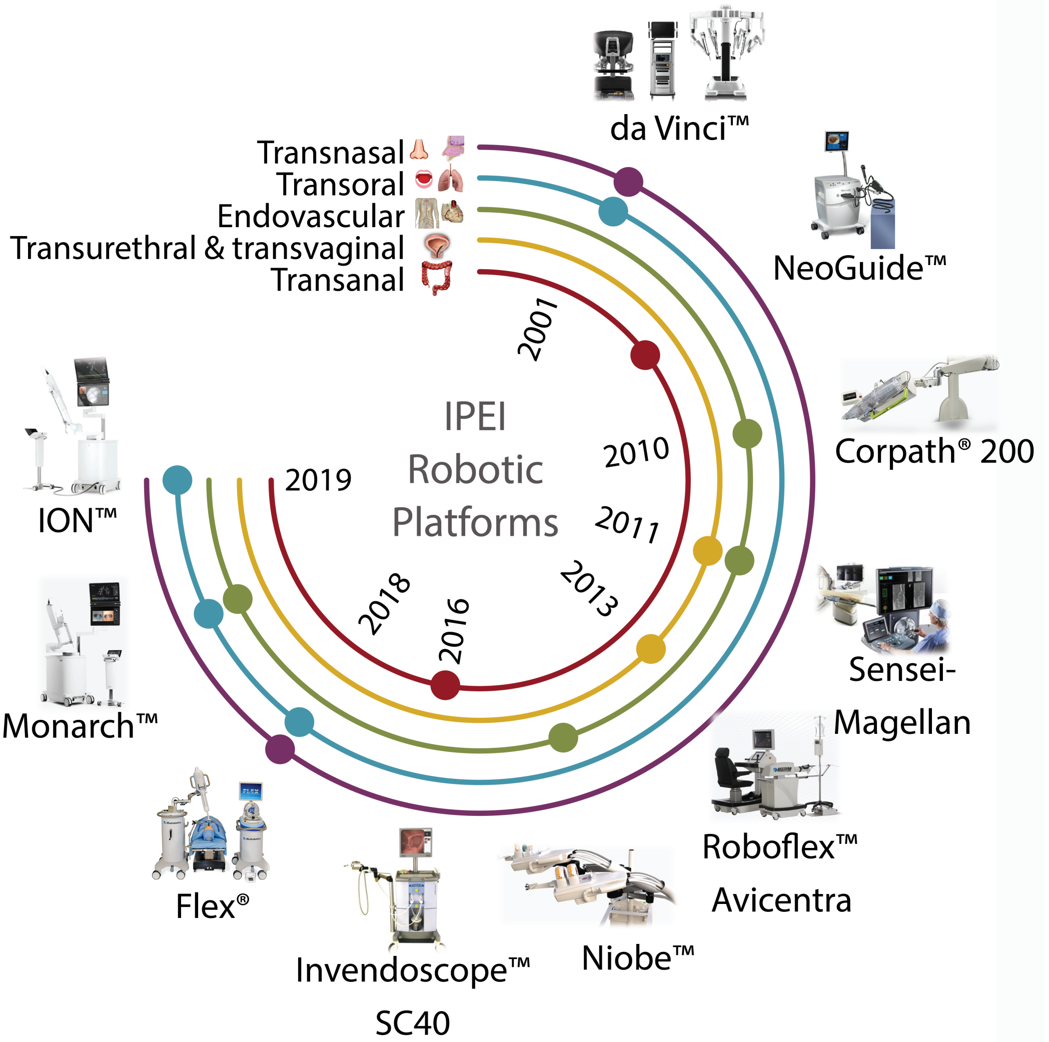


Fig. 2.2 Selection of some commercial robotic systems for IP. For endovascular interventions: Corpath™ system (Corindus, Waltham, USA) and Niobe™ system (Stereotaxis, St. Louis, USA), Sensei–Magellan (Hansen Medical, Mountain View, USA) and Monarch system (Auris Health, Redwood, USA); For transurethral and transvaginal procedures: Roboflex™ (ELMED, Ankara, Turkey) and Sensei–Magellan (Hansen Medical, Mountain View, USA); For gastrointestinal transanal procedures: Invendoscope™ (Invendo Medical, Weinheim, Germany) and Aer-O-Scope (GI View Ltd, Ramat Gan, Israel); For transnasal procedures: da Vinci (Intuitive Surgical, Sunnyvale, USA) and Flex® (Medrobotics, Raynham, USA) For bronchoscopic transoral intervention: Monarch system (Auris Health, Redwood, USA), ION™ (Intuitive Surgical, Sunnyvale, USA), da Vinci (Intuitive Surgical, Sunnyvale, USA) and Flex® (Medrobotics, Raynham, USA) are used.

Robotic colonoscopy has been investigated to simplify the use of flexible endoscopes, reduce procedure time, and improve the overall outcome [83]. Some cost-efficient solutions have shown advantages in reducing pain, sedation requirements, and disposability [28]. These platforms have a self-propelling semi-autonomous or teleoperated navigation system.

Several robotic colonoscopy platforms have received clearance to enter the market, including the NeoGuide Endoscopy System (NeoGuide Endoscopy System Inc., Los Gatos, USA) [90], the Invendoscope™ E210 (Invendo Medical GmbH, Weinheim, Germany), the Aer-O-Scope System (GI View Ltd., Ramat Gan, Israel)[91], the ColonoSight (Stryker GI Ltd., Haifa, Israel) [92], and the Endotics System (ERA Endoscopy Srl, Pisa, Italy) [93]. However, the NeoGuide Endoscopy system and the ColonoSight are no longer commercially available. The Neoguide system is a cable-driven system consisting of 16 independent segments, each with 2 DoFs, position sensors at the tip to obtain the insertion depth, and real-time 3D mapping of the colon. In contrast, the Invendoscope™ E210 is a single-use, pressure-driven colonoscope that grows from the tip using a double layer of an inverted sleeve, reducing the forces applied to the colonic wall. The device has a working channel with electrohydraulic actuation at the tip.

The ColonoSight is composed of a reusable endoscope wrapped with a disposable sheath to prevent infection. The locomotion is provided by the air inflated inside the sleeve that covers an inner tube. The tip consists of a bendable section with two working channels. The Aer-O-Scope is a disposable self-steering and propelling endoscope that uses electro-pneumatic actuation through two sealed balloons. Recent proof-of-concept of the device showed successful caecum intubation with no need for sedation [91]. The Endotic System uses a remotely controlled disposable colonoscope that mimics inchworm locomotion.

2.4.3 Transurethral and transvaginal IP

Transurethral and transvaginal interventions are commonly used in urological surgeries, including bladder cancer resection, radical prostatectomy, partial cystectomy, and nephrectomy [94, 95]. These interventions involve the use of an endoscopic device to intentionally puncture a viscera, such as the ureter or urinary bladder, to access the abdominal cavity and perform intra-abdominal operations [96].

However, the widespread adoption of transurethral and transvaginal access for urological applications is limited by several challenges. One of the major challenges is the lack of dedicated, specially designed instruments, resulting in limited distal dexterity, tool accuracy, and depth perception [97, 95]. These limitations lead to under-resection of tumors and difficulty in enucleating tissue, especially with minimal tilting of the rigid tools and the urethral anatomy. These challenges motivate research in robot-assisted techniques for urological interventions [94].

In 2008, the Sensei-Magellan system (Hansen Medical, Mountain View, USA), originally designed for cardiology and angiography, was used for Robotic Flexible Ureteroscopy (fURS) [98]. Since 2010, ELMED (Ankara, Turkey) has developed the Roboflex™ Avicenna system for fURS, which directly drives the endoscope and an arm enabling rotation by a joystick. This system offers improved movement precision and better ergonomics compared to traditional flexible ureteroscopy [99].

2.4.4 Transoral IP

Transoral Endoscopy (TOE) is a standard diagnostic method used to examine the esophagus, stomach, and proximal duodenum, employing varying lengths of flexible endoscopes such as gastroscopes (925mm–1.1 m), duodenoscopes (approximately 1.25 m), and enteroscopes (1.52- 2.2 m) [100]. The success of TOE relies heavily on the technical and decision-making skills of the operator, with a steep learning curve [101]. Laryngeal lesions are conventionally treated using standard endoscopic surgical approaches involving a laryngoscope, microscope, and laser [102]. However, these approaches require the surgeon to work within the limits of the laryngoscope, resulting in line-of-sight observation limitations [102]. TOE is also used for bronchoscopy procedures to reach the lungs farther down the airways, employing bronchoscopes [103]. However, the average diagnostic yield remains low due to limited local view in the peripheral airways [75]. ElectroMagnetic (EM) navigation was introduced to guide the bronchoscope through the peripheral pulmonary lesions, but it lacked direct visualization of the airways, thereby motivating the need for robotic assistance [104].

Currently available robotic systems for TOE include the EASE system (EndoMaster Pte, Singapore) and EndoSamurai™ (Olympus Medical Systems Corp., Tokyo, Japan). The EASE system is a teleoperated device that remotely controls the endoscopic medical arms. The EndoSamurai™ system consists of instruments mounted at the end of the endoscope for submucosal dissection procedures. Several other robotic systems, currently in the early development phase, are reviewed in [101].

Commercially available systems for laryngeal procedures include the dVSS (Intuitive Surgical, Sunnyvale, USA) and the Flex Robotic System (Medrobotics, Raynham, USA) [105]. The Flex robotic system includes a rigid endoscope controlled through a computer interface, with two external channels for flexible instruments.

Monarch™ (Auris Health Inc, Redwood, USA) is pioneering robotic endoscopy in bronchoscopy procedures. The platform consists of an outer sheath, an inner bronchoscope with 4DoFs steering control, electromagnetic navigation guidance, and continuous peripheral visualization [104]. Another robotic platform called ION™ Endoluminal System by Intuitive Surgical includes an articulated, flexible catheter with shape sensing capabilities, providing positional and shape feedback along with a video probe for live visualization while driving the catheter [104].

2.4.5 Transnasal IP

Transnasal IP procedures have been explored for a variety of targets, including sinuses, skull base, and upper airways. The difficulty in monitoring the progression of sinus diseases, obtaining a biopsy, and facilitating intervention in the frontal and maxillary sinuses without visible scarring or bone scaffolding obliterating are some of the challenges faced by this approach [106]. Conventionally, a flexible endoscope is used in clinical practice. Skull base surgeries are performed through transnasal access, with a typical target being the removal of pituitary gland tumors through a transsphenoidal approach [107, 108]. The endoscopic approach for these surgeries is limited by restricted access, manual manipulation of interventional tools near susceptible anatomy, and lack of distal dexterity.

The upper airways and throat is another interventional target that is accessible through the transnasal approach. Transnasal Endoscopy (TNE) is conducted using an ultrathin endoscope with a shaft diameter of 6 mm inserted through the nasal passage. Once the instrument is beyond the upper esophageal sphincter, endoscopy is conducted in the standard fashion. However, TNE has some technical limitations, such as a smaller working channel, which can result in limited suction and the availability of fewer endoscopic accessories [109].

Robotic systems that have been mentioned in transoral approaches, such as the dVSS and Flex® Robotic System, have also been used in transnasal interventions [105]. The Flex® Robotic System is an operator-controlled flexible endoscope system designed primarily for ear, nose, and throat procedures that include a steerable endoscope and computer-assisted controllers, with two external channels for the use of compatible 3.5 mm flexible instruments. However, specific robotic systems with appropriate ergonomics and dimensions suited for transnasal passage are still under development [105].

2.5 Conclusions

In this chapter, we have discussed the regulatory considerations necessary for approving medical robotic systems, highlighting recent efforts in regulating autonomous systems in safety-critical areas such as surgery. One of the prime concerns for validating medical robotics is safety, and a framework needs to be developed to track the source of error when the robot functionality differs from expected. However, these regulatory standards are not fully developed for robot-assisted intervention, and the introduction of levels of autonomy could support this development by facilitating system verification and validation with improved risk management.

To this end, we have introduced the LoA in robot-assisted MIS and extended the framework to IP. We believe that the most promising features of upcoming IP robotic systems are autonomy, as they provide the ability to perceive, analyze, plan, and take actions automatically. It should be noted that most autonomous features in development are still in the research phase and slowly entering the market. Thus, we have provided an overview of robotic advancements for various IP routes such as transoral, transanal, transnasal, transurethral and transvaginal, and endovascular. Robotic systems for IP have shown to improve precision, dexterity, and visualization during surgical procedures. They have also helped in reducing patient trauma, hospital stays, and overall costs. These advancements hold great promise for improving the clinical outcomes and revolutionizing the field of robot-assisted MIS.

Contributions of this chapter

1. LoA for IP: These LoA provide a framework for developing and evaluating autonomous robotic systems for IP. Advances in robotic technology are enabling higher LoA in these procedures, which can lead to increased safety, efficiency, and accuracy.

2. Overview of commercially existing robotic systems for various IP routes that include transoral, transanal, transnasal, transurethral and transvaginal, and endovascular, with their limitations

Publications linked to this chapter

1. Ameya Pore, Zhen Li, Diego Dall'Alba, Albert Hernansanz, Elena De Momi, Arianna Menciassi, Alicia Casals, Jenny Denkelman, Paolo Fiorini and Emmanuel Vander Poorten."Autonomous Navigation for Robot-assisted Intraluminal and Endovascular Procedures: A Systematic Review", accepted in Transactions on Robotics (T-RO).

Chapter 3

Robotic reinforcement learning for Surgery: Background

The emergence of deep learning has had a profound impact on numerous areas of machine learning, resulting in substantial improvements in tasks such as object detection, speech recognition, and language translation [110]. The central attribute of deep learning is the ability of Deep Neural Network (DNN) to automatically generate compact, low-dimensional representations (features) of high-dimensional data, such as images, text, and audio. This capability has similarly facilitated progress in RL by enabling the resolution of previously intractable decision-making problems in high-dimensional state and action spaces [111], commonly referred to as DRL. DRL has demonstrated early success in tasks such as playing video games, chess, and Go at superhuman levels [112, 41, 113].

Surgical robotics presents a promising domain for evaluating DRL algorithms due to its ability to combine learning with simultaneous perception and movement in a safety-critical scenario. The adoption of learning-based methods in surgical robotics research is particularly appealing as it can empower robots to operate in less structured environments, handle unknown objects, and learn state representations appropriate for multiple tasks. The field of robot learning for surgical tasks is situated at the confluence of machine learning, robotics, and surgery, with RL serving as a seminal mathematical framework for experience-driven autonomous learning. By enabling surgical robots to learn from their experiences and optimize their actions, robotic learning has the potential to greatly enhance the precision, efficiency, and safety of surgical interventions. Recent research has shown great potential in robotic control for surgical tasks, such as tissue manipulation and dissection [49, 114].

However, the application of RL to surgical robotics poses several challenges. The cost of a robot is a significant factor, and numerous design decisions must be made when setting up the algorithm and the robot. RL algorithms require autonomous collection of experience by the robot, which raises questions about how learning should be initialized, how to prevent unsafe behavior, and how to define the goal or reward.

In this chapter, we provide a comprehensive overview of the RL formulation and a description of commonly used DRL algorithms. We present the challenges associated with robotic learning for surgery and the different ways used to mitigate these problems.

3.1 Reinforcement learning formulation

The various IP subtasks that we consider in this thesis, i.e. navigation and tissue manipulation, are formulated independently in an RL framework where an autonomous agent perceives a state s_t from its environment at timestep t. The agent then selects an action a_t in state s_t to interact with the environment. Based on the current state and the chosen action, the environment and the agent transition to a new state s_{t+1}. The state is a sufficient statistic of the environment and thus contains all the necessary information for the agent to make the best action selection, which can involve components of the agent, such as the position of its sensors and actuators.

In this thesis, we experiment with different state-spaces such as the endoscopic image, which provides the information about the luminal environment and kinematic state information that provides positional information of the objects present in the surgical scene.

The optimal sequence of actions is determined by the rewards provided by the environment, which encodes the task objective. Each time the agent and environment transition to a new state, the environment provides a scalar reward r_{t+1} to the agent as feedback. The agent's objective is to learn the navigation and tissue manipulation policy (control strategy) π that maximizes the expected return. Given a state, a policy returns an action to perform. An optimal policy is any policy that maximizes the expected return in the environment. However, RL presents a challenge because the agent must learn the consequences of actions in the environment through trial and error, as it does not have access to a model of the state transition dynamics, unlike in optimal control. Every interaction with the environment provides information that the agent uses to update its knowledge.

3.1.1 Markov Decision Process

Formally, RL can be described as a MDP, which consists of the following components:

- A set of states S, along with a distribution of starting states $p(s_0)$.
- A set of actions A
- Transition dynamics $T(s_{(t+1)}|s_t, a_t)$ that map a state-action pair at time t onto a distribution of states at time $t+1$.
- An immediate/instantaneous reward function $R(s_t, a_t, s_{t+1})$.
- A discount factor $\gamma \epsilon [0, 1]$, where lower values place more emphasis on immediate rewards.

The term policy (π) generally refers to a function that maps states to a probability distribution over actions, i.e., $\pi : S \rightarrow p(A = a|S)$. In an episodic MDP, where the state is reset after each episode of length T, the sequence of states, actions and rewards in an episode is known as a trajectory or rollout of the policy. Each rollout of a policy results in a return $R = \sum_{t=0}^{T-1} \gamma^t r_{t+1}$, which is the cumulative discounted reward obtained by the agent.

The objective of RL is to find an optimal policy, π^*, that maximizes the expected return from all states:

$$\pi^* = \underset{\pi}{\operatorname{argmax}} \quad E[R|\pi] \tag{3.1}$$

Here, the expectation E is taken over all possible trajectories that can be generated by following the policy π.

Two main approaches exist to tackle RL problems: one is based on value functions while the other relies on policy search [111].

3.1.2 Value Functions

Value function methods are based on estimating the state-value function $V^{\pi}(s)$, which is the expected return when starting in state s and following π, henceforth:

$$V^{\pi}(s) = E[R|s, \pi] \tag{3.2}$$

The optimal policy π^* has a corresponding state-value function $V^*(s)$ and vice-versa, the optimal state-value function can be defined as

$$V^*(s) = \max_{\pi} V^{\pi}(s), s \in S \tag{3.3}$$

When $V^*(s)$ available, the optimal policy could be retrieved by choosing among all actions available at s_t and picking the action a that maximizes

$$E_{s_{t+1} \sim T(s_{t+1}|s_t, a)}[V^*(s_{t+1})]$$

In the RL setting, the transition dynamics T are unavailable [111, 115]. Therefore, a state action value function $Q_{\pi}(s, a)$ is constructed as an alternative to V^{π}, such that the initial action a is provided, and π is followed from the succeeding state onwards:

$$Q^{\pi}(s, a) = E[R|s, a, \pi] \tag{3.4}$$

The best policy, given $Q^{\pi}(s, a)$, can be found by choosing a greedily at every state: $\underset{a}{\operatorname{argmax}} Q^{\pi}(s, a)$ Under this policy, $V^{\pi}(s)$ can be defined by maximizing $Q^{\pi}(s, a) : V^{\pi}(s) = max_a Q^{\pi}(s, a)$.

The function Q^{π} is learned by exploiting the Markov property and defining the function as a Bellman equation [116, 39]. The equation takes on a recursive form and is given by:

$$Q^{\pi}(s_t,a_t) = E_{s_{t+1}}[r_{t+1} + \gamma Q^{\pi}(s_{t+1}, \pi(st+1))]. \tag{3.5}$$

This allows for bootstrapping, whereby the current values of the estimate of Q^{π} can be used to improve the estimate.

Temporal Difference (TD) methods simulate only one step under current state instead of reaching the terminal state [39]. The simplest TD method, TD(0) is:

$$V(s_t) = V(s_t) + \alpha(R_{t+1} + \gamma V(s_{t+1}) - V(s_t)) \tag{3.6}$$

where $R_{t+1} + \gamma V(s_{t+1})$ is the estimated value at $t+1$, called TD target, and $R_{t+1} + \gamma V(s_{t+1}) - V(s_t)$ is called TD error. In TD(0), after every step, value function is updated with the value of the next state and reward obtained along the way.

Agents that use TD error to update the value function can be trained using methods such as Q-learning [117]. The Q-learning update equation is:

$$Q(s_t,a_t) = Q(s_t,a_t) + \alpha(R_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s_t,a_t)) \tag{3.7}$$

where $\alpha \in (0,1]$ is the step size.

Deep Q-Learning Conventional Q-learning updates Q-values using the TD method and stores them in a Q-table. However, this approach is not feasible for problems with large state and action spaces. To address this issue, Mnih *et al.* proposed the DQN [112], which combines Q-learning with DNN.

DQN uses a DNN to extract low-level features from raw images of Atari games and approximate the action-value function without requiring any domain knowledge. The hidden layer of DQN comprises three convolutional layers and a fully connected layer, as shown in Fig. 3.1. The output layer produces the Q-value of each action. At time step t, the approximated value of DQN is given by:

$$y = R + \gamma \max_a Q(s_{t+1},a;\theta) \tag{3.8}$$

Here, θ represents the parameters of the DQN, which are updated by minimizing the Mean Square Error (MSE) between the approximated and real Q-values.

Another major value-function based method relies on learning the advantage function $A_{\pi}(s,a)$, which differs from Q_{π} by only considering the relative values of state-action pairs [39]. The advantage function represents the advantage of taking a particular action in a given state over taking the average action, and is calculated using the relationship $A_{\pi} = Q_{\pi} - V_{\pi}$. Learning relative values instead of absolute values is similar to removing the baseline or

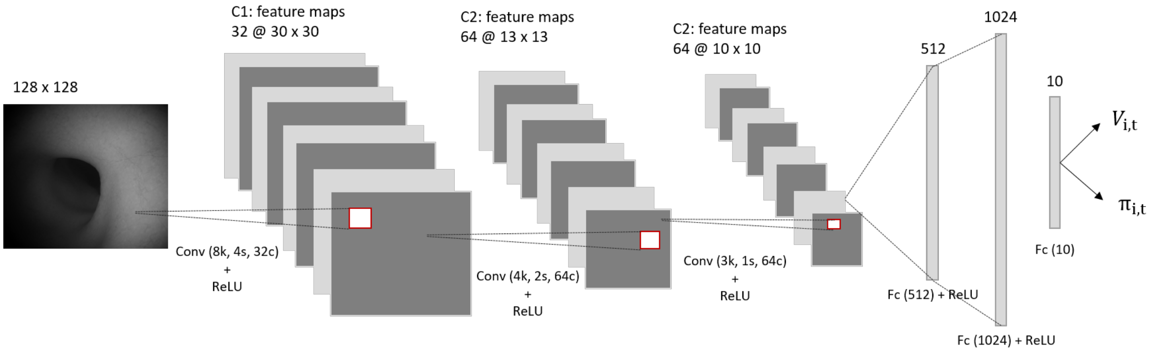


Fig. 3.1 Schematic illustration of a CNN based DQN. The input to the network consists of a 128×128 image, followed by three convolutional layers, and a fully connected layer. The output of the fully connected layer is a probability distribution that indicates the action to take and the corresponding value of the state.

average level of a signal. This approach simplifies the learning process by focusing on learning which actions are better than others, rather than trying to learn the actual return of taking an action. Advantages updates have been incorporated into various DRL algorithms, including asynchronous RL [118] and high-dimensional continuous control [119].

3.1.3 Policy search

Policy search is a popular approach for solving RL problems that directly search for an optimal policy π^* without maintaining a value function model. Typically, a parameterized policy π_θ is chosen and its parameters are updated to maximize the expected return $E[R|\theta]$ using either gradient-based or gradient-free optimization methods [120].

Gradient-free policy search methods require a heuristic search across a predefined class of models to find better policies [111]. One of the advantages of gradient-free policy search is that they can optimize non-differentiable policies. DNN have been successfully trained to encode policies using both gradient-free [121, 122] and gradient-based [123, 119, 124, 125] methods. Although gradient-free optimization is effective in covering low-dimensional parameter spaces, gradient-based training is still the method of choice for most DRL algorithms because of its sample efficiency, despite some successes in applying gradient-free methods to large networks [126].

When constructing a policy directly, it is common to output parameters for a probability distribution function. For continuous actions, the output can be the mean and standard deviations of Gaussian distributions, while for discrete actions, the output can be the individual probabilities of a multinomial distribution. This results in a stochastic policy from which actions can be directly sampled:

$$\pi_\theta(a|s) = P(a|s,\theta) \tag{3.9}$$

Policy gradient Policy gradient methods leverage gradients to improve the parameterized policy. However, to calculate the expected return, we need to average over all plausible trajectories generated by the current policy parameterization. Therefore, the performance measure of policy π_θ can be defined as the expected return, and expressed as:

$$J(\theta) = V_{\pi_\theta}(s) = E_{\pi_\theta(s)}\left[\sum_a Q(s,a)\pi_\theta(a|s)\right] \tag{3.10}$$

The policy gradient theorem can then be used to obtain the equation for policy optimization by differentiating V with respect to θ [127]. This yields:

$$\nabla_\theta J(\theta) = E_{\pi_\theta(s)}\left[\sum_a Q(s,a)\nabla_\theta \pi_\theta(a|s)\right] \tag{3.11}$$

The policy parameters can then be updated by adding the scaled gradient to the current policy parameters, expressed as:

$$\theta = \theta + \alpha \nabla_\theta J \tag{3.12}$$

where α is the step size.

The Reinforce algorithm [128] is a conventional policy gradient RL algorithm that updates the policy using estimated cumulative returns from sampled trajectories. The expected value of the sample's gradients is an unbiased estimate of the actual gradient, which is the reason for using Reinforce. The most commonly used variant of Reinforce is the form with a baseline, which helps to reduce the variance generated when estimating the gradient.

$$\nabla_\theta J(\theta) = E_{\pi_\theta(s)}\left[\nabla_\theta log \pi_\theta(a|s)(Q(s,a) - b)\right] \tag{3.13}$$

where b is usually a learned state-value function independent of a.

Actor-Critic methods

Actor-critic methods are a family of RL algorithms that simultaneously learn a policy and a value function, with the value function being used to evaluate the policy [39]. The actor generates policies, selects actions and interacts with the environment, and is updated using gradients calculated from equations 3.11 and 3.12. The critic evaluates the value function of the actor's policy at each time step. Various measures can be used to evaluate the actor's policy, including the action-value function $Q(s,a)$, state value function $V(s)$, or advantage function $A(s,a)$.

Advantage Actor-Critic (A3C): Conventional policy gradient algorithms are usually updated in an on-policy manner, where the policy being used to update the value function and policy is the same policy that the agent is currently following to select actions. This approach results in slow policy convergence and low data efficiency [39]. To speed up the training of actor-critic methods, Mnih *et al.* [118] proposed the asynchronous advantage actor-critic (A3C) algorithm, which collects data asynchronously. A3C uses N threads to interact with the environment simultaneously. As each thread has a different environment setting, the interaction trajectories obtained from each thread are not the same, which speeds up sample collection. After the samples are collected, each thread completes the training independently and updates the global model parameters asynchronously. The policy gradient equation is updated as follows:

$$\nabla_\theta J(\theta) = E_{\pi_\theta(s)}\left[\nabla_\theta log\pi_\theta(a|s)(Q(s,a) - V(s))\right] \tag{3.14}$$

where $(Q(s,a) - V(s))$ is the advantage function.

Trust Region Policy Optimization: In policy optimization equations (3.12), the step size parameter α plays a crucial role in determining the speed of policy convergence. However, selecting an appropriate value for α is challenging as an unsuitable value can lead to unstable or even deteriorated policies. One effective solution to this problem is to employ trust regions that restrict optimization steps to a region where the approximation of the true cost function is still valid. By constraining the updated policies to be close to previous policies, trust regions decrease the chance of bad updates and improve the monotonicity of policy performance. One prominent algorithm that uses trust regions for policy optimization is the Trust Region Policy Optimization (Trust region policy optimization (TRPO)), which is relatively robust and applicable to high-dimensional input domains [123].

To implement trust region constraints, TRPO optimizes an advantage estimate with a quadratic approximation of the Kullback–Leibler (KL) divergence. Although TRPO can be employed as a pure policy gradient method with a simple baseline, the Generalized Advantage Estimation (GAE) method by Schulman *et al.* [119] introduced several advanced variance reduction baselines to improve performance. The combination of TRPO and GAE has become a state-of-the-art RL technique in continuous control. However, TRPO is limited by the need to compute second-order gradients.

A newer algorithm called PPO performs unconstrained optimization, requiring only first-order gradient information [129]. PPO has two primary variants: an adaptive penalty on the KL divergence and a heuristic clipped objective that is independent of the KL divergence. PPO iteratively collects new observations and improves the policy while approximating the value function. The update function for the PPO policy is given by the following equation:

$$L(s,a,\theta_k,\theta) = min\left(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}\hat{A}^{\pi_{\theta_k}}(s,a), \quad g(\epsilon, \hat{A}^{\pi_{\theta_k}}(s,a))\right) \tag{3.15}$$

where θ_k are the parameters of the old policy, and g is defined as:

$$g(\epsilon, \hat{A}) = \begin{cases} (1+\epsilon)\hat{A}_t, & \hat{A}_t \geq 0 \\ (1-\epsilon)\hat{A}_t, & \hat{A}_t < 0 \end{cases} \tag{3.16}$$

In order to prevent unstable policy updates that can occur when the update step size or the policy ratio is too large, the PPO algorithm introduces two measures in the objective function. Firstly, the ratio $\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}$ is constrained to lie within the range of $[1-\epsilon, 1+\epsilon]$. This ensures that the magnitude of policy updates is limited, thereby preventing significant fluctuations. Secondly, the objective function includes a min function that selects the lower value of the two results.

Deterministic Policy Gradients: Deterministic Policy Gradient (DPG)s [130] are a recent development in the context of actor-critic algorithms, extending the standard policy gradient theorem from stochastic [128] to deterministic policies. DPGs offer a significant advantage over stochastic policy gradients in that they only integrate over the state space, rather than both state and action spaces, thus requiring fewer samples in problems with large action spaces. In their initial work on DPGs, Silver *et al.* [130] introduced an off-policy actor-critic algorithm that vastly improved upon a stochastic policy gradient equivalent in high-dimensional continuous control problems. In the context of a deterministic policy, the selected action $a = \mu(s)$ remains unaltered for a given state s. The deterministic policy gradient is defined as:

$$\nabla_\theta J(\theta) = E\left[\nabla_a Q_\mu(s,a) \nabla_\theta \mu(s)|_{a=\mu_\theta(s)}\right] \tag{3.17}$$

where μ is the deterministic policy function, and Q_μ is the corresponding action-value function. The DDPG algorithm [124] is a specific implementation of DPGs that learns deterministic policies and expands them into continuous action spaces through an actor-critic architecture.

3.2 Challenges of robotic learning in IP

In contrast to supervised learning where considerable progress has been made in large-scale, easy deployment of RL is not yet applicable to surgical robotics. Surgical robotics as a RL domain differs considerably from most well-studied RL benchmark problems.

State Estimation: One major hurdle is that the states and actions of most robotic systems are inherently continuous (i.e. a set of possible states that can take on infinitely many values within a given range), and therefore, we must determine how finely to represent them. We must decide how fine grained the control is that we require over the robot, whether we employ discretization or function approximation, and what time step we establish. Robotic platforms for IP pose a particular challenge due to their high-dimensional state and action space arising from the large number of DoFs and image-based input. As the dimensionality increases, the number of required data and computations to cover the entire state-action space grows exponentially and becomes infeasible, widely known as the “Curse of Dimensionality” [131]. Furthermore, it is often not practical to assume that the true state is entirely observable and free of noise in robotics RL [132]. These are due to factors such as wear and tear of robotic hardware, delays in sensing and external conditions such as temperature and lighting. Therefore, robotic RL is commonly treated as partially observable, and the learning system must estimate the true state while maintaining uncertainty estimates [39].

Model errors and under-modeling: Obtaining experience on a robotic system is tedious, expensive and often hard to reproduce. Even getting to the same initial state is almost impossible for the surgical robotic system due to the dynamic surgical environment. In order to learn within a reasonable time frame, suitable approximations of state, policy, value function, and/or system dynamics need to be introduced. While real-world experience is costly, it usually cannot be replaced by learning in simulations alone. In analytical or learned models of the system, even small modeling errors can accumulate to substantially different behavior, at least for highly dynamic tasks. This problem may be inevitable due to the uncertainty and non-stationarity of the true system dynamics. Hence, algorithms need to be robust with respect to models that do not capture all the details of the real system. Some commonly used approaches deal with this problem by incorporating model uncertainty with artificial noise or carefully choosing reward functions to discourage controllers that generate frequencies that might excite unmodeled dynamics [133].

Reward Design: Another challenge commonly faced in robot RL is the generation of appropriate reward functions that guide the learning system quickly to success, which are needed to cope with the cost of real-world experience. These reward function must be programmed, or otherwise learned by the robot [45]. In a surgical scenario, assigning a score to quantify how well a task was completed can be a challenging perceptual problem. In most of our case studies, we sidestep this difficulty by instrumenting the environment with additional sensors that provide reward information. For example using an EM tracker to know the groundtruth 3-D position of the region of interest in an IP. Robotic RL approaches often need more physically motivated reward-shaping based on continuous values and consider multi-objective reward functions like minimizing the motor torques while achieving a task. A learning problem is potentially difficult if the reward is sparse, there are significant delays between an action and the associated significant reward, or if the reward is not smooth (i.e., very small changes to the policy lead to a drastically different outcome).

3.3 Tractability Through Representation and Prior Knowledge

Much of the success of RL methods in robotics is largely due to the use of approximate representations. However, the ability to use such representations effectively is tightly linked to the optimization framework employed. For instance, state-action discretization is a popular way of reducing the dimensionality of states or actions, and can enhance both policy search and value function-based methods [134].

Value function-based methods require a function approximator that is both accurate and robust, in order to capture the value function with sufficient precision while maintaining stability during learning. On the other hand, policy search methods require a policy representation that controls the complexity of representable policies to enhance learning speed. Prior knowledge can also play a vital role in guiding the learning process. It can be incorporated in the form of initial policies, demonstrations, initial models, predefined task structure, or constraints on the policy such as torque limits or ordering constraints of the policy parameters [135]. These approaches can significantly reduce the search space, thus speeding up the learning process. Providing a (partially) successful initial policy enables RL methods to concentrate on promising regions in the value function or in policy space.

3.3.1 Smart state-action discretization

Discretization of the state and action spaces is a common method for reducing the dimensionality of the problem [134]. Several studies have developed manual discretizations for basic tasks to be learned on real robots [136, 137]. For low-dimensional tasks, discretizations can be generated straightforwardly by dividing each dimension into a set of regions. However, the key challenge lies in determining the appropriate number of regions for each dimension, which enables the system to achieve optimal performance while still learning efficiently [134].

State spaces can also be constructed based on different features, such as positions, shapes, and colors, for learning object affordances, where both the discrete sets and the mapping from sensor values to discrete values need to be designed. Instead of manually specifying the discretizations, adaptive methods can be employed to construct them during the learning process [138].

3.3.2 Learning from Demonstration (LfD)

Incorporating demonstrations into RL can offer several advantages. First, it provides a supervised training set that specifies what actions to perform in states that are encountered. Such data can be useful for biasing policy action selection [139, 134]. Second, the use of demonstrations or a hand-crafted initial policy eliminates the need for global exploration of the policy or state-space of the RL problem. This enables the agent to improve by locally optimizing a policy, knowing what states are important, making local optimization methods feasible [135].

However, the discovery of optimal solutions in the learning framework is limited to local optima near the demonstrated behavior. This reliance on demonstrations for a favorable starting point suggests that reducing the necessity for extensive global exploration can facilitate the learning process. The primary objective of LfD is to enable the robot to acquire and replicate the demonstrated behavior while also generalizing it to novel and unfamiliar scenarios [115].

In the LfD approach, a dataset of demonstrations $D = (\tau_i, s_i, r_i), \quad i = 1,...N$ is collected, which includes a tuple of trajectories τ, state observation s, and possibly reward signals r. The dataset can be collected either offline or online. A common optimization-based approach learns a policy π^* using the collected dataset D, such that

$$\pi^* = \underset{\pi}{\operatorname{argmin}} \quad D(q(\phi), p(\phi|\pi)) \tag{3.18}$$

where $q(\phi)$ is the distribution of features induced by the expert's policy, $p(\phi|\pi)$ is the distribution of features induced by the learner following π, and $D(q,p)$ is a similarity measure between q and p.

Current LfD methods are commonly categorized into three types: Behavioral cloning (BC), Inverse Reinforcement Learning (IRL), and adversarial imitation learning [135]. These methods aim to learn the policy from the demonstrations provided by the expert, either by directly copying the expert's behavior, by inferring the expert's underlying reward function, or by generating a policy that is indistinguishable from the expert's behavior, respectively.

Behavioral Cloning (BC) methods aim to learn a policy directly from demonstrated data by mapping the state to the control input through standard supervised learning methods [111]. To achieve this, BC methods require a surrogate loss function that quantifies the difference between the demonstrated behavior and the learned policy. One of the simplest options for a surrogate loss function is the *L1-Loss function* given by:

$$l_{BC}(q,p) = \sum_i |q_i - p_i| \tag{3.19}$$

The method of Inverse Reinforcement Learning (IRL) aims to recover the reward function that represents the expert's intention by utilizing the optimality of the expert's teaching information, and subsequently utilizes RL methods to derive the final control strategy based on the recovered reward function [140]. This can be beneficial when the reward function is the most parsimonious way to describe the desired behavior. The goal of IRL is to obtain the unique solution for the unknown reward function $R(\tau)$ from the expert's trajectories. However, since a policy can be optimal for multiple reward functions, determining the reward function is an "ill-posed" problem. To address this issue, various studies have proposed additional objective functions that can be optimized, such as the margin between the optimal and other policies [141, 142] or maximizing entropy [143]. The effectiveness of the strategy generated by IRL may be reduced in an environment that differs significantly from the teaching environment, since IRL assumes the optimality of the expert's teaching information [140].

In contrast to BC and IRL methods that only rely on expert teaching information to learn strategies, adversarial imitation learning methods use Generative Adversarial Network (GAN) to further optimize the learning strategies [144]. The GAIL technique is a type of adversarial imitation learning that confronts the generated trajectory with the expert teaching trajectory, distinguishes the expert trajectory from the imitator trajectory using a classifier, and iteratively trains the system to minimize the distribution distance between the two trajectories for complete operational imitation. In this thesis, we employ GAIL to learn surgical gestures from expert surgeons, as described in Chapter 8.

Generative Adversarial Imitation Learning (GAIL)

GAIL is an imitation learning algorithm based on GAN [144]. The GAIL approach involves a policy generator G_ϕ and a discriminator $D_{\phi'}$, where ϕ and ϕ' denote the parameters associated with each network. The generator produces exploration trajectories that are evaluated by the discriminator using a surrogate function to measure the similarity between the generated and expert policies. This similarity metrics acts as a reward proxy for the RL step. Unlike IRL methods, GAIL directly generates policies rather than the reward function. The discriminator is trained to minimize the loss function:

$$L_{GAIL} = E_{\tau_\phi}[\log(D_{\phi'}(s_t,a))] + E_{\tau_E}[\log(1 - D_{\phi'}(s_t,a))] \tag{3.20}$$

where τ_ϕ and τ_E are the trajectories generated by G_ϕ and the expert trajectories, respectively. The policy generator is often implemented using methods based on stochastic policy, such as PPO, due to its stable and diverse trajectory generation [129]. PPO generates a wide sampling range of trajectories that serve as a good training set for the discriminator in GAIL.

3.3.3 Prior Knowledge Through Task Structuring

Task decomposition into hierarchical subtasks or into a sequence of increasingly difficult tasks can facilitate learning by providing prior knowledge to the learning process. Decomposition into subtasks has several advantages such as: (i) subtasks are easier and faster to learn than learning an overall control policy; (ii) modular behaviors are easier to interpret and can be adapted to similar tasks [145].

Task decomposition: The concept of breaking down a task into smaller subtasks has been widely studied in the literature [146, 147], where these subtasks are then coordinated to produce a complex behavior. Recently, HRL has emerged as a RL setting where multiple agents can be trained at various levels of temporal abstraction [148] and can learn different subtasks using an end-to-end training paradigm. In HRL, agents are trained such that the low-level agent encodes primitive motor skills while the higher-level policy selects the low-level agents to complete a task [149, 150]. Similarly, Beyret *et al.* [151] proposed an explainable HRL method for a robotic manipulation task that uses HER as a high-level agent to select goals that are provided as input to the low-level policy. However, in these works, hierarchical policies are learned end-to-end, leading to instability and sample inefficiency due to the lower-level policy changing under a non-stationary high-level policy. To address this instability limitation, we propose to train the low-level policy independently from the high-level policy to efficiently learn a robotic pick and place task, described in Appendix 1.

3.4 Conclusions

Robotic RL for surgical applications presents a number of challenges that are unique to robotics setting. Some common challenges include high-dimensional continuous state and action space, partial observability, noise and high sample cost, described in Sec. 3.2.

In this chapter, we provide an overview of commonly used RL methods, and how they can be approached in the surgical robotics context. For robotic IP, we employ policy gradient methods such as PPO to automate the navigation task, described in Chapter 5.

Exploration can pose a major limitation in robotic RL. One way to guide the policy towards an optimal policy and speed up the learning process is by leveraging effective representations, incorporating prior knowledge and task structuring. Effective representation can be extracted by simplifying the RL problem by discretization or reducing the dimensionality of state and action space. We discuss LfD methods such as BC, IRL and GAIL. In the context of IP, we use GAIL to learn human-like trajectories to automate the tissue manipulation gesture described in Chapter 8. Finally, we developed a task decomposition method based on HRL to learn a complex pick and place task, presented in Appendix 1.

Learning in real surgical environment requires various kinds of environmental instrumentation and human intervention in order to define the reward functions, the reset between trials, obtain ground truth state and monitor hardware status and ensure safety [152]. Overcoming these challenges in a scalable way requires designing robotic systems that possess three capabilities: they are able to (1) learn from their own raw sensory inputs, (2) assign rewards to their own trials without hand-designed perception systems or instrumentation, and (3) learn continuously in non-episodic settings without requiring human intervention to manually reset the environment. A system with these capabilities can autonomously collect large amounts of real world data – typically crucial for effective generalization – without significant instrumentation in each training environment. Such a system would also bring us significantly closer to the goal of embodied learning-based systems that improve continuously through their own real-world experience.

Besides the challenges associated to the surgical and the robotics domain, current DRL methods themselves face several drawbacks such as sample inefficiency and safety that limit the ability to train robots directly in real world. These DRL related challenged are discussed in Chapter 4.

Contributions of this chapter

1. Overview of RL literature and commonly used DRL methods.

2. Challenges associated with robotic learning and steps to mitigate them

Part I

Endoscopic Navigation

Chapter 4

Motion planning for Intraluminal Procedures

Autonomous IP robotics presents a challenging domain for Motion Planning (MP) algorithms due to the restrictions imposed by patient safety and complex anatomical environments involved. Navigation in such procedures necessitates effective perception, precise control, and reliable modeling. MP has been an extensively studied field for navigation tasks since the 1980s, with applications in mobile platforms and robotic manipulators in both indoor and outdoor industrial settings. In MP, the robot's geometrical dimensions and kinematic constraints are considered to obtain a feasible path solution that avoids collisions. The relationship between the robot's configuration and task spaces is described by its kinematics. The configuration space, denoted as $\mathcal{C}$, refers to all possible robot configurations, while the task space, denoted as $\mathcal{T}$, is the workspace accessible by the robot for each specific configuration $\mathbf{q}$. The robot kinematics can be expressed in a general form as

$$\mathcal{T} = f(\mathbf{q}) \quad \mathbf{q} \in \mathcal{C} \tag{4.1}$$

This chapter introduces the survey analysis for MP methods in Sec. 4.1. It presents the taxonomy and classification of MP algorithms for IP in Sec. 4.2. Each algorithm is discussed in detail, including its strengths, weaknesses, and validation. The chapter also provides a comparative analysis of the different MP algorithms, based on their efficiency, scalability, and applicability to different types of IP. One of the key to successful clinical translation of autonomous motion control is to test the methods in simulated environments. Hence, Sec. 4.3 provides an overview of existing virtual and physical simulation platform for common IP. Finally, we provide the limitation of MP methods in Sec. 4.4.

4.1 Literature Survey Methodology

A systematic analysis was conducted, following the PRISMA methodology [153], to survey the developments of automation and MP in IP.

Search method

To conduct the analysis, we used three digital libraries, namely, `Google Scholar`, `Scopus`, and `IEEE Xplore`. Search queries were programmatically generated using a search term matrix that was designed to generalize the term "motion planning for intervention". The search terms were combined with logical operators AND and OR to cover a large search space in sufficient detail. For example, a search query would take the format of “planning AND *vascular AND catheter”. A total of 520 entries were obtained from the search, covering various research topics, application scenarios, and clinical devices.

The search results were automatically managed, retrieved and checked for duplicates using a python library called Pybliometrics [154]. The list of references was saved as a .csv file and manually evaluated according to the inclusion criteria. All items that did not meet the inclusion criteria were excluded. The cutoff date for the earliest work included in the analysis was 2005, and the latest work was from July 2022. The search methodology is presented in detail in Sec. 4.1. Fig. 4.1 provides an overview of all the search terms used in the study and the flow of the conducted review.

Selection criteria

The research work selection criteria were as follows:

1. Only continuum robots were considered for IP, while capsule mobile robots were excluded (e.g., [155, 156]).
2. Studies on low-level controllers, such as force control, position control, impedance control, and similar, were excluded.
3. Only full papers written in English were considered. Extended abstracts reporting preliminary findings were omitted.
4. Transluminal procedures that require incisions, such as hydrocephalus ventricles, were excluded.

Post processing and analysis

The search script retrieved a total of 11,404 references, which were imported into a spreadsheet software and screened for inclusion by the authors based on the predefined criteria. The titles

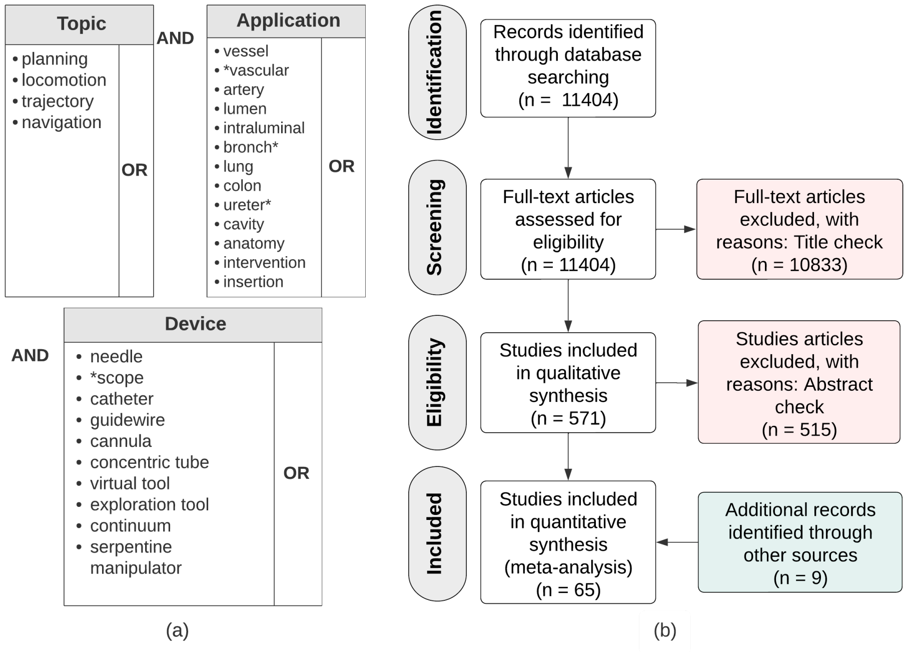

Topic
• planning
• locomotion
• trajectory
• navigation
OR
AND
Application
• vessel
• *vascular
• artery
• lumen
• intraluminal
• bronch*
• lung
• colon
• ureter*
• cavity
• anatomy
• intervention
• insertion
OR
AND
Device
• needle
• *scope
• catheter
• guidewire
• cannula
• concentric tube
• virtual tool
• exploration tool
• continuum
• serpentine manipulator
OR
(a)
Identification
Records identified through database searching (n = 11404)
Screening
Full-text articles assessed for eligibility (n = 11404)
Full-text articles excluded, with reasons: Title check (n = 10833)
Eligibility
Studies included in qualitative synthesis (n = 571)
Studies articles excluded, with reasons: Abstract check (n = 515)
Included
Studies included in quantitative synthesis (meta-analysis) (n = 65)
Additional records identified through other sources (n = 9)
(b)

Fig. 4.1 (a) Search matrix used for the survey (b) PRISMA flow diagram summarizing how the systematic review was conducted.

of each reference were evaluated, and those meeting the inclusion criteria were shortlisted while those that did not were excluded. If the inclusion status was unclear from the title, the paper was included to avoid the inadvertent omission of potentially relevant material. The systematic review's process is summarized in the PRISMA flow diagram in Fig. 4.1b. After the title check, 10,833 references were excluded, and an additional 515 references were excluded after the abstract check. Nine references were included manually. Ultimately, this process yielded a list of 65 references.

The outcomes of the various studies were classified based on several criteria, as shown in Fig. 4.2, including the targeted procedure, the LoA, the MP method, the validation, and the environment's dynamics. The MP methods are categorized into subgroups presented in Fig. 4.3 for an in-depth analysis. The summary of the state-of-the-art on IP MP publications is presented in Table 4.2, and its development is shown in Fig. 4.4a. In addition to the MP approach, the distribution of targeted IP procedures is highlighted in Fig. 4.4b. Furthermore, Table 4.2 indicates that some studies involved intraoperative path replanning with a dynamic environment (last column).

4.2 Taxonomy on motion planning for IP navigation

The taxonomic classification of MP methods for IP is depicted in Fig. 4.3. The methods can be categorized into four sub-groups, which are node-based, sampling-based, optimization-based,

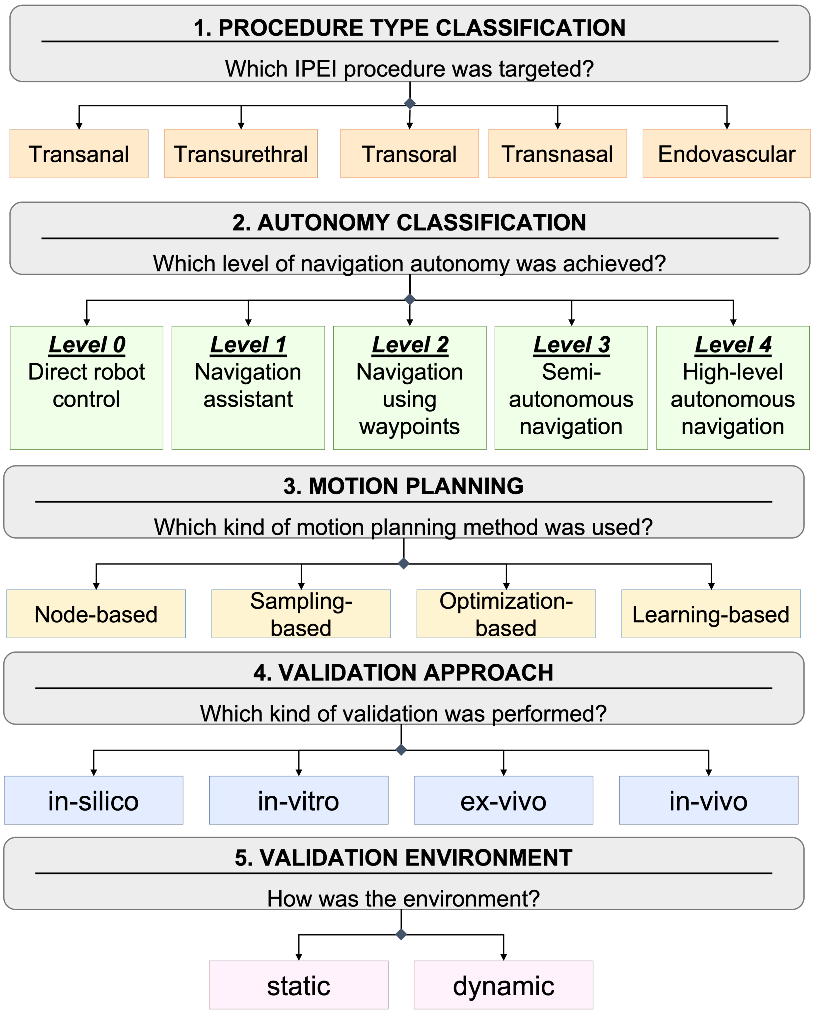

1. PROCEDURE TYPE CLASSIFICATION
Which IPEI procedure was targeted?
Transanal
Transurethral
Transoral
Transnasal
Endovascular
2. AUTONOMY CLASSIFICATION
Which level of navigation autonomy was achieved?
Level 0 Direct robot control
Level 1 Navigation assistant
Level 2 Navigation using waypoints
Level 3 Semi-autonomous navigation
Level 4 High-level autonomous navigation
3. MOTION PLANNING
Which kind of motion planning method was used?
Node-based
Sampling-based
Optimization-based
Learning-based
4. VALIDATION APPROACH
Which kind of validation was performed?
in-silico
in-vitro
ex-vivo
in-vivo
5. VALIDATION ENVIRONMENT
How was the environment?
static
dynamic

Fig. 4.2 Schematics of the analysis carried out for each paper. These criteria include the targeted procedure, the level of autonomy, the motion planning method, the validation and the dynamics of the environment.

and learning-based techniques. These sub-groups are derived from the general taxonomy of path planning [1] for robots, as presented in [157]. Node-based, also known as graph-based, algorithms utilize a tree structure and a graph-searching strategy to find a collision-free path. Sampling-based algorithms, on the other hand, construct a tree structure based on random samples in the configuration workspace, ensuring that the path found is collision-free and compatible with the robot's motion capabilities. Optimization-based algorithms formulate

[1] Path planning is the problem of finding a collision-free path (a list of discrete setpoints or a continuous curve) from one configuration (or state) to another. A path is defined in the workspace. Conversely, a trajectory is a path with a specification of the time at which each configuration is achieved. It can be defined in the joint space as well. Trajectory planning takes into consideration robot kinematics and dynamics, while path planning considers only geometric constraints. Both path planning and trajectory planning can be viewed as a subclass of motion planning. Motion planning is the general term for finding a collision-free motion for the robot system from one configuration (or state) to another. It defines the change of state at any instant.

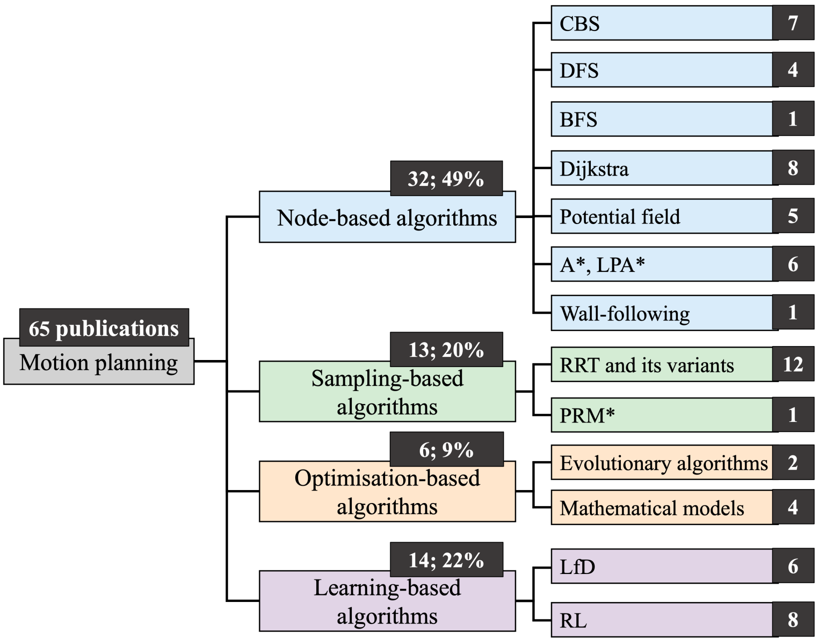


Fig. 4.3 Classification of IP motion planning methods for continuum robots found in literature

the MP problem as a mathematical problem, where an objective function is minimized or maximized with respect to constraints, and an optimal solution is obtained through a solver. Finally, learning-based methods employ a MDP to learn a goal-directed policy based on a reward function. A brief introduction of specific path planning methods in a general field is provided in Table 4.1.

4.2.1 Node-based algorithms

Node-based algorithms utilize a graph based approach to represent an environment map where each node represents a configuration of the robot or agent and each edge represents a feasible transition between two configurations [157]. The algorithms build the graph by sampling the configuration space of the robot or agent and connecting nodes that are close enough to each other. The resulting graph is then searched to find a path from the start node to the goal node using various search algorithms such as Centerline-based Structure (CBS), Depth First Search (DFS), Breadth First Search (BFS), Dijkstra, potential field, A*, Lifelong Planning A* (LPA*), and wall-following, as illustrated in Fig. 4.3. Table 4.2 provides a summary of various MP works for IP that employ node-based methods.

Centerline-based Structure: Geiger *et al.* extracts the 3D skeleton for bronchoscopy planning by computing the skeleton of the segmented structure and then converting this skeleton into a hierarchical tree model of connected branches [158]. The generation of virtual bronchoscopy is often limited because of insufficient peripheral bronchi identification resulting

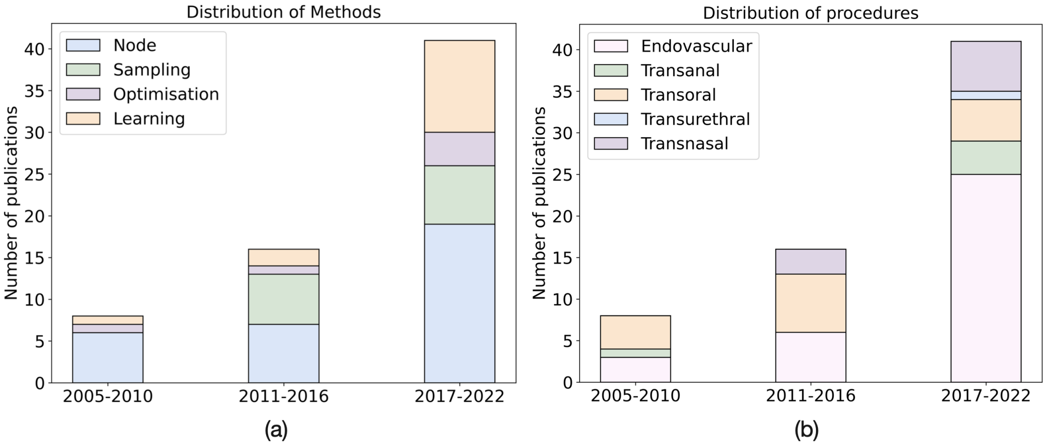


Fig. 4.4 Chronological development of endoluminal navigation. (a) Motion-planning approaches (b) The targeted IP procedures. Until 2010, the majority of studies have implemented node-based and sampling algorithms for MP. While lately, with the exponential increase in computational resources, the field is transitioning towards learning-based methods.

from the limitation in CT airway resolution [158]. Geiger *et al.* overcomes this limitation by using peripheral arteries as surrogates. Sánchez *et al.* [159] obtains the skeleton of the bronchial anatomy via the fast marching method firstly and then defines the skeleton branching points as a binary tree (B-tree). Sánchez' study labels the skeleton branching points according to their heading direction (1-left, 2-right) and gives a path corresponding to a sequence of nodes traversing the B-tree. Intraoperatively, a geometry likelihood map is used to match the current exploration to the path planned preoperatively. The airway centerlines serve as the natural pathways for navigating through the airway tree. They are represented by a discrete set of airway branches in [71]. Starting with each target Region of Interest (ROI) associated airway route, the method from Khare *et al.* [71] automatically derives a navigation plan that consists of natural bronchoscope maneuvers abiding by the rotate-bend-advance paradigm learned by physicians during their training. This work is evaluated both in phantoms and in a human study. The reported results show that it achieves a success rate of 97% in airway route navigation and a mean guidance time per diagnostic site of 52 s.

Wang *et al.* developed a method to build a navigation information tree based on the vasculature's centerline for catheterization [160]. The authors made a tree structure assuming the vascular system was rigid and interrogated the tree to find the nearest node during intraoperative navigation. The navigation experiments were carried out on a resin vessel phantom. Another study proposed a 3D vasculature's centerline extraction approach via a Voronoi diagram [161]. It treated the centerlines as the minimal action paths on the Voronoi diagrams inside the vascular model surface. The experimental results show that the approach can extract the centerlines of the vessel model. Further Zheng *et al.* [14] firstly propose to extract the preoperative 3D skeleton via a parallel thinning algorithm for medical axis extraction [162]. Secondly, they propose to use a graph matching method to establish the correspondence between the 3D preoperative and 2D intraoperative skeletons, extracted from 2D intraoperative fluoroscopic images. However, the proposed graph matching is sensitive to topology variance and transformation in the sagittal and transverse planes. Some recent work on transnasal exploration, by [163], proposed central path extraction algorithm based on pre-planning for the roaming area.

Nevertheless, a common disadvantage of work available in the literature describing this approach is that they focus on constructing an information structure, but path exploration inside the information structure is not mentioned [158, 159, 71, 160, 161, 14]. Specifically, the tree structure is built, but the path solution is not generated autonomously through a graph search strategy, especially when there are multiple path solutions simultaneously.

Depth First Search: As an extended method to travel the tree formed in [71], the work by Zang *et al.* implements a route search strategy of DFS for an integrated endobronchial ultrasound bronchoscope, exploring a graph by expanding the most promising node along the depth [72], [164]. In another study by Gibbs *et al.*, a DFS to view sites is regarded as the first phase search, followed by a second search focusing on a ROI localization phase and a final refinement to adjust the viewing directions of the bronchoscope [165]. A DFS approach is also developed in Huang *et al.* for endovascular interventions [166]. Instead of considering path length as node weights in the typical DFS approach, this work defines the node weights as an experience value set by doctors.

The search time and the planned path are significantly dependent on the order of nodes in that same graph layer. Even though a DFS approach can search for a feasible path by first exploring the graph along with the depth, it does not ensure that the first path found is the optimal path.

Breadth First Search: This algorithm was employed in [167] for a magnetically-actuated catheter to find a path reaching the target along vascular centerline points. However, the BFS algorithm would take much more time to find a solution in a complex vascular environment with multi-branches.

Dijkstra: A graph structure based on vasculature's centerlines that are determined using a volume growing and a wavefront technique is designed by Schafer *et al.* in [168]. The optimal path is then determined using the shortest path algorithms from Dijkstra. However, Schafer *et al.* assume that the centerline points are input as an ordered set, which would be a strict assumption. Moreover, they only report the scenario of a single lumen without branches, which does not reflect the advantages of the Dijkstra algorithm. A similar method but in a backward direction is presented by Egger *et al.* [169]. This work determines an initial path by Dijkstra. Users define initial and destination points. After that, the initial path is aligned with the blood vessel, resulting in the vasculature's centerline. However, this methodology is not fully autonomous, and it involves manually tuned parameters. Another work extracts the centerline and places a series of guiding circular workspaces along the navigation path that are perpendicular to the path [170]. The circular planes jointly form a safe cylindrical path from the start to the target. The Dijkstra algorithm is implemented to find the minimal cumulative cost set of voxels within the airway tree for bronchoscope navigation [171, 172] and find the shortest path along vasculature's centerlines [173], [174, 175].

In comparison to DFS, Dijkstra algorithm continuously monitors and verifies the cost until it reaches the intended target, leading to a greater probability of obtaining an improved solution. Nevertheless, these researches still focus on tracking anatomical centerlines that are difficult to follow precisely and often not desirable. This is because aligning the instrument tip with the centerline may require excessive forces at more proximal points along the instrument's body where contact with the anatomy occurs.

Potential field: The work by Rosell *et al.* [176] computes the potential field based on the L1 distance to obstacles. It is used to search a path by wavefront propagation for bronchoscopy. Rosell's approach considers the geometry and kinematic constraints while selecting the best motion according to a cost function. Yang *et al.* [177] extract centerlines via a distance field method, establish and navigate the tree after that. However, the authors only considered the curvature constraint at 180° turns along vasculature's centerlines and assumed that all the path points have the same Y coordinate. Martin *et al.* [178] employ a potential field approach by defining an attractive force from the endoluminal image center to the colon center. Starting from endoluminal images, the colon is detected via the FAST edge feature detector, and the center of this area is computed. A linear translation between the colon center and the image center is reconstructed and regarded as the linear motion of the colonoscope tip. This work is validated both in the synthetic colon and pig colon (*in-vivo*). A similar approach is followed by Zhang *et al.* where a robotic endoscope platform is employed to bring surgical instruments at the target site [179]. Girerd *et al.* [180] use a 3D point cloud representation of a tubular structure and compute a repulsive force to ensure that the concentric tube needle tip remains inside the contour.

The Potential field has an advantage in local planning by maintaining the center of the image close to the center of the cross-section of the lumen or the vessels. Nevertheless, it only considers a short-term benefit rather than global optimality during this local planning and might get stuck in a local minimum during global path planning.

A* and Lifelong Planning A*: He *et al.* [181] compute and optimize endoscopic paths using the A* algorithm. The effectiveness of the preoperatively planned path is verified by an automatic virtual nasal endoscopy browsing experiment. Ciobirca *et al.* search shortest airway paths through voxels of a bronchus model using the A* algorithm [182]. They claimed that this method could potentially improve the diagnostic success rate with a system for tracking the bronchoscope during a real procedure. However, this statement has not been validated yet. Some studies proposed a path planning method for Concentric Tube Robot (CTR)s in brain surgery. The authors of these studies build a nearest-neighbor graph and use LPA* algorithm for efficient replanning to optimize the insertion pose [183, 184]. Compared to A*, LPA* [185] can reuse information from previous searches to accelerate future ones. Ravigopal *et al.* proposed a modified hybrid A* search algorithm to navigate a tendon-actuated coaxially aligned steerable guidewire robot along a pre-computed path in 2D vasculature phantoms under C-arm fluoroscopic guidance [186]. Recently, Huang *et al.* showed colon navigation using real-time heuristic searching method, called Learning real-time A* (LRTA*) [187]. LRTA* with designed directional heuristic evaluation shows efficient performance in colon exploration compared to BFS and DFS. Directional biasing avoids the need for unnecessary searches by constraining the next state based on local trends.

A* and LPA* use heuristic information to reach the goal. They can converge very fast and ensures optimality as well. A* is commonly used for static environments, while LPA* can adapt to changes in the environment. Nevertheless, the speed execution of A* and LPA* depends on the accuracy of the heuristic information.

Table 4.1 Background of path-planning methods.

No.	Path Planning	Description
1.	**Node-based**	
a.	Centerline-based Structure (CBS)	This method is long-established to keep the tip of the instruments away from the walls [188]. A tree structure is built from the anatomical information of the lumen, where each node contains the information of the lumen centerline position and the corresponding lumen radius.
b.	Depth First Search (DFS)	DFS algorithm traverses a graph by exploring as far as possible along each branch before backtracking [189]
c.	Breadth First Search (BFS)	BFS algorithm [190] starts at the tree root and explores the k-nearest neighbor nodes at the present depth before moving on to the nodes at the next depth level.
d.	Dijkstra	The Dijkstra algorithm [191] is an algorithm for finding the shortest paths between nodes in a graph. It is also called Shortest Path First (SPF) algorithm. The Dijkstra algorithm explores a graph by expanding the node with minimal cost.
e.	Potential field	Artificial potential field algorithms [192] define a potential field in free space and treat the robot as a particle that reacts to forces due to these fields. The potential function is composed of an attractive and repulsive force, representing the different influences from the target and obstacles, respectively.

f.	A* & Lifelong Planning A* (LPA*)	A-star [193] is an extension of the Dijkstra algorithm, which reduces the total number of states by introducing heuristic information that estimates the cost from the current state to the goal state.
g.	Wall-following	Wall-following algorithms move parallel and keep a certain distance from the wall according to the feedback received from sensors.
2.	**Sampling-based**	
a.	Rapidly-exploring Random Tree (RRT)	RRT [194] and its derivatives are widely used sampling-based methods. These methods randomly sample in the configuration space or workspace to generate new tree vertices and connect the collision-free vertices as tree edges. In addition, these methods can consider the kinematic constraints (i.e., curvature limitations) during MP.
b.	Probabilistic RoadMap* (PRM*)	A probabilistic roadmap is a network graph of possible paths in a given map based on free and occupied spaces [195, 196]. PRM* takes random samples from the robot's configuration space, tests them for whether they are in the free space, and uses a local planner to attempt to connect these configurations to other nearby configurations. Then, the starting and goal configurations are added in, and a graph search algorithm is applied to the resulting graph to determine a path between these two configurations.
3.	**Optimization-based**	
a.	Mathematical Model	MP can be formulated as a path optimization problem with constraints on the robot model, such as its kinematic model [197].
b.	Evolutionary algorithms	Evolutionary algorithms use bio-inspiration to find approximate solutions to difficult optimization problems. [197]. Ant Colony Optimization (ACO) is one of the population-based metaheuristic algorithms [198]. Artificial ants incrementally build solutions biased by a pheromone model, i.e. a set of parameters associated with graph components (either nodes or edges) whose values are modified at runtime by the ants.
4.	**Learning-based**	
a.	Learning from Demonstrations (LfD)	LfD is the paradigm where an agent acquires new skills by learning to imitate an expert. LfD approach is compelling when ideal behavior cannot be easily scripted, nor defined easily as an optimization problem, but can be demonstrated [199].
b.	Reinforcement Learning (RL)	In RL, an agent learns to maximise a specific reward signal through trial and error interaction with the environment by taking actions and observing the reward [39].

Wall-following: The study in [200] uses a wall-following algorithm to assist catheter navigation. Fagogenis *et al.* [200] employ haptic vision to accomplish wall-following inside the blood-filled heart for a catheter. The wall-following algorithm could be considered an efficient navigation approach if there are few feasible routes to reach the target state. Otherwise, the solution of a wall-following algorithm cannot ensure optimality.

4.2.2 Sampling-based algorithms

As observable in Table 4.2, different works, in the context of MP for IP, exploit sampling-based methods. As schematized in Fig. 4.3, algorithms based on Rapidly-exploring Random Tree (RRT) and its variants and Probabilistic RoadMap* (PRM*) have been proposed.

Rapidly-exploring Random Tree and its variants: The RRT algorithm and its variants have been extensively studied for their ability to generate optimal paths in virtual bronchoscopy simulators. Aguilar *et al.* [201, 202] compared several RRT-based algorithms, including RRT, RRT-connect, dynamic-domain RRT, and RRT-Connect with dynamic-domain, and found that RRT-Connect with Dynamic Domain is the optimal method that requires the least number of samples and computational time for finding the solution path.

Fellmann *et al.* [203] used a collision-free path via RRT as a baseline and evaluated four trajectory generation strategies: asynchronous/synchronous point-to-point, rotation before translation, and translation before rotation. They measured the path length and number of collisions and found that synchronous point-to-point is the best strategy inside narrow and straight nasal passages. However, this strategy may not be feasible for larger distances between intermediate configurations.

Kuntz *et al.* [204] proposed a three-step planning approach using a RRT-based algorithm for a transoral lung system comprising a bronchoscope, a CTR, and a bevel-tip needle. Their approach considers the needle's steering ability during path planning and respects the maximum needle steering curvature. The authors demonstrated that the motion planner could find a motion plan for 36% of the lung nodule locations in 1 second, for 70% in 60 seconds, and for 75% in 1 hour, after performing 50 trials on 50 lung nodule locations. The time to find a motion plan depends on the steering capability and the target location.

The study in [205] implements an improved RRT algorithm for cerebrovascular intervention. The expansion direction of the random tree is a compromise between the new randomly sampled node and the target. This strategy can improve the convergence speed of the algorithm. However, their work did not take into account any constraints of the several constraints imposed by the catheters such as kinematic limitations and dynamic capabilities. Alterovitz *et al.* [206] proposed a Rapidly-exploring RoadMap (RRM) method that initially explores the configuration space like RRT. Once a path is found, RRM uses a user-specified parameter to weigh whether to explore further or to refine the explored space by adding edges to the current roadmap to find higher-quality paths in the explored space. Their method is presented for CTRs in a tubular environment with protrusions as bronchus. Some studies develop the RRM method and improve it with more accurate mechanics-based models in a skull base surgery scenario and static lung bronchial tubes for CTRs respectively [207], [208]. In Torres *et al.* [207], the planner required 1077 s to get a motion plan that avoids bones, critical blood vessels and healthy brain tissue on the way to the skull base tumor. The same authors extend the previous studies in [209] by proposing a modified Rapidly-exploring Random Graph (RRG) method that computes motion plans at interactive rates. If progress towards the goal can be made by following the roadmap, an A* graph search is used to find the shortest motion plan to the node nearest the goal. This work improves the computation cost and allows replanning when the robot tip position changes. However, generating such a roadmap requires an extensive amount of computation. Therefore, the method could behave well in a static environment but not in deformable lumens.

Fauser *et al.* use the formulation of RRT-connect (or bi-directional RRT, Bi-RRT) introduced earlier by them [210] to solve a common MP problem for instruments that follow curvature constrained trajectories [211]. In [212], Fauser *et al.* implement the RRT-connect algorithm for a catheter in a 3D static aorta model, under the allowed maximal curvature $0.1\,\mathrm{mm}^{-1}$. Further extension of this work proposes path replanning [213]. Replanning from different robot position states along the initial path takes place at $0.6(1)\,\mathrm{s}$ from the start at the descending aorta to the goal in the left ventricle.

Probabilistic RoadMap*: Kuntz *et al.* propose a method based on a combination of a PRM* method and local optimization to plan motions in a point cloud representation of a nasal cavity anatomy [214]. Point cloud representations, if updated, can accommodate for the anatomy's intraoperative changes (i.e., before/after blockage removal). After performing 100 trials, the success rate in the upper airway, colon and skull base scenarios were found to be 98%, 99% and 98%, respectively. The limitation is that the anatomy model is only updated within the visible region of the endoscope, while deformations of the rest of the anatomy are not considered. If tissue deformation is negligible, this planning method could be used for intraoperative planning. Otherwise, the deformations of the overall model must be considered beforehand.

4.2.3 Optimization-based algorithm

MP can be formulated as an optimization problem and solved by numerical solvers [197]. Moreover, these methods can be programmed to consider also the robotic kinematics.

Mathematical model: An optimization-based planning algorithm that optimizes the insertion length and orientation angle of each tube for a CTR with five tubes is proposed by Lyons *et al.* [215]. Firstly, the authors formulate the MP problem as a non-linear constrained optimization problem. Secondly, the constraint is moved to the objective function, and the problem is converted to a series of unconstrained optimization problems. Lastly, the optimal solution is found using the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm [216] and Armijo's Rule [217]. The robot kinematics is modeled using a physically-based simulation that incorporates beam mechanics. This work is evaluated in simulation on a patient's lung anatomy. However, the computational time of the proposed method is high, which restricts the possibility of applying it to real-time scenarios. Moreover, the authors manually define the skeleton and treat the structure as a rigid body, confining its applicability.

Qi *et al.* [218] present an inverse kinematics MP approach for continuum robots, which formulates the problem as an optimization based on the backbone curve method. The kinematic model is built on the premise of piecewise constant curvature, with the technique minimizing the distance to the vasculature's centerline under kinematic constraints. The algorithm can be executed in real-time with an average advancement speed of 0.4mm/s. The method considers the constrained optimization problem within the overall configuration space, avoiding the reduction of the search space. However, the approach processes the optimization problem independently at each step without considering long-term cumulative cost, resulting in optimal inverse kinematics that may not be globally optimal.

Guo *et al.*[219] proposed a directional modeling approach for a teleoperated catheter and a hybrid evaluation function to determine the optimal trajectory. The effectiveness of this method was evaluated through wall-hit experiments, and the response time of obstacle avoidance with and without path planning was compared. However, this approach relies on exhaustive enumeration to find the optimal solution, which sacrifices time to obtain a solution. On the other hand, Abah *et al.*[220] formulated the path planning problem as a nonlinear least-squares problem, aiming to minimize the passive deflection of the steerable catheter by matching the shape of the steerable segment to that of the cerebrovascular. Nonetheless, the vascular centerline may not be the optimal reference route.

Evolutionary algorithms: Gao *et al.*[221] proposed an improved version of the ACO method that considers factors such as catheter diameter, vascular length, diameter, curvature, and torsion to plan an optimal vascular path. However, the method's computational time ranged from 2 s to 30 s, with an average of 12.32 s, which limits its application in real-time scenarios. Li *et al.*[222] proposed a fast path planning approach that satisfies the steerable catheter curvature constraint using a local Genetic Algorithm (GA) optimization. The method achieves a low computational time cost of 0.191 ± 0.102s while ensuring compliance with the robot curvature constraint.

4.2.4 Learning-based algorithms

Learning-based methods have emerged as a promising solution for real-time MP. These methods rely on statistical techniques like Hidden Markov models (HMMs), DNN and dynamical models to map perceptual and behavioral spaces. Among learning methods, two sub-fields have been identified as relevant to the context of this research: LfD and RL approaches.

Learning from Demonstrations (LfD): Rafii-Tari *et al.* present a system for human-robot collaboration in catheterization using hierarchical HMMs [223]. The system decomposes catheterization into a sequence of motion primitives, which are modeled as HMMs and learned using a LfD approach. A high-level HMM is also learned to sequence these motion primitives. The authors justify the use of hierarchical HMMs due to their ability to handle spatial and temporal variability across multiple demonstrations, while also allowing for motion sequence generation and recognition of new motions. In another study by the same authors, a semi-automated approach for navigation is proposed where guidewire manipulation is controlled manually, and catheter motion is automated by the robot [224]. Catheter motion is modeled using a Gaussian Mixture Model (GMM) to create a representation of temporally aligned phase data generated from demonstrations. Chi *et al.* expand on this work by demonstrating subject-specific variability among type I aortic arches by incorporating anatomical information obtained from preoperative image data [225]. Expectation maximization is used to perform maximum-likelihood estimation to learn model parameters in all of the above methods.

Chi *et al.* propose a LfD method based on Dynamical Movement Primitives (DMPs) [13]. DMPs are compact representations for motion primitives formed by a set of dynamic system equations [226]. The study uses DMPs to avoid unwanted contacts between the catheter tip and the vessel wall. DMPs are trained from human demonstrations and used to generate motion trajectories for the proposed robotic catheterization platform. The proposed methods can adapt to different flow simulations, vascular models, and catheterization tasks.

In a recent continuation of their prior study, Chi *et al.* improve the RL part by including model-free GAIL loss that learns from multiple demonstrations of an expert [227]. In this work, the catheterization policies adapt to the real-world setup and successfully imitate the task despite unknown simulated parameters such as blood flow and tissue-tool interaction. Zhao *et al.* propose a GAN framework by combining CNN and Long Short Term Memory (LSTM) [228] to estimate suitable manipulation actions for catheterization. The DNNs are trained using expert demonstration data and evaluated in a phantom with a grayscale camera simulating X-ray imaging.

Table 4.2 Summary of motion planning methods for IPEI navigation

Ref.	Procedure	Method	Robot	Kinematic	Validation	Environment
Level 1: Navigation assistant						
Node-based:						
[158] Geiger 2005	Transoral	CBS	Bronchoscope	N	in-silico (3D pulmonary vessels)	Static
[168] Schafer 2007	Endovascular	Dijkstra	Guidewire	N	in-vitro (3D cardiovascular)	Static
[169] Egger 2007	Endovascular	Dijkstra	Catheter	N	in-silico (3D aorta)	Static
[171] Gibbs 2007	Transoral	Dijkstra	Bronchoscope	N	in-silico (3D bronchus)	Static
[172] Gibbs 2008	Transoral	Dijkstra	Bronchoscope	N	in-silico (3D bronchus)	Static
[170] Liu 2010	Endovascular	Dijkstra	Catheter	N	in-vitro (3D aorta)	Static
[160] Wang 2011	Endovascular	CBS	Catheter	N	in-vitro (3D resin vessel)	Static
[166] Huang 2011	Endovascular	DFS	Guidewire	N	in-silico (3D aorta)	Static
[176] Rosell 2012	Transoral	Potential field	Bronchoscope	Y	in-silico (3D tracheobronchial)	Static
[165] Gibbs 2013	Transoral	DFS	Bronchoscope	Y	in-vivo (3D bronchus)	Dynamic
[161] Yang 2014	Endovascular	CBS	Guidewire	N	in-silico (3D aorta)	Static
[71] Khare 2015	Transoral	CBS	Bronchoscope	Y	in-vivo (3D bronchus)	Dynamic
[159] Sánchez 2016	Transoral	CBS	Bronchoscope	N	in-silico (3D bronchus)	Static
[14] Zheng 2018	Endovascular	CBS	Catheter	N	in-vitro (3D aorta)	Dynamic
[182] Ciobirca 2018	Transoral	A*	Bronchoscope	N	in-silico (3D bronchus)	Static
[183] Niyaz 2018	Transnasal	LPA*	Concentric tube robot	Y	in-silico (3D nasal cavity)	Static
[184] Niyaz 2019	Transnasal	LPA*	Concentric tube robot	Y	in-silico (3D nasal cavity)	Static
[72] Zang 2019	Transoral	DFS	Bronchoscope	Y	in-vivo (3D bronchus)	Dynamic
[177] Yang 2019	Transurethral	Potential field	Ureteroscope	Y	in-silico (3D ureter)	Static
[200] Fagogenis 2019	Endovascular	wall-following	Concentric tube robot	Y	in-vivo (3D cardiovascular)	Dynamic
[181] He 2020	Transnasal	A*	Endoscope	N	in-silico (3D nasal cavity)	Static
[164] Zang 2021	Transoral	DFS	Bronchoscope	Y	in-vivo (3D bronchus)	Static
[187] Huang 2021	Transanal	LRPA*	Colonoscope	Y	in-vitro (2D colon)	Dynamic
[163] Wang 2021	Transnasal	CBS	Endoscope	Y	in-silico (3D nasal cavity)	Static
[186] Ravigopal 2021	Endovascular	Hybrid A*	Robotic guidewire	Y	in-vitro (2D vessel)	Static
sampling-based:						
[206] Alterovitz 2011	Transoral	RRM	Concentric tube robot	Y	in-silico (3D bronchus)	Static
[207] Torres 2011	Transnasal	RRM	Concentric tube robot	Y	in-silico (3D nasal cavity)	Static
[208] Torres 2012	Transoral	RRM	Concentric tube robot	Y	in-silico (3D bronchus)	Static
[209] Torres 2014	Transnasal	RRG	Concentric tube robot	Y	in-silico (3D nasal cavity)	Static
[203] Fellmann 2015	Transnasal	RRT	Concentric tube robot	Y	in-silico (3D nasal cavity)	Static
[204] Kuntz 2015	Transoral	RRT	Steerable needle	Y	in-silico (3D bronchus)	Static
[201] Aguilar 2017	Transoral	bi-RRT	Bronchoscope	Y	in-silico (3D bronchus)	Static
[202] Aguilar 2017	Transoral	bi-RRT	Bronchoscope	Y	in-silico (3D bronchus)	Static
[211] Fauser 2018	Endovascular	bi-RRT	Catheter	Y	in-silico (3D vena cava)	Static
[212] Fauser 2019a	Endovascular	bi-RRT	Steerable guidewire	Y	in-silico (3D aorta)	Static
[213] Fauser 2019b	Endovascular	bi-RRT	Steerable guidewire	Y	in-silico (3D aorta)	Static
[214] Kuntz 2019	Transnasal	PRM*	Concentric tube robot	Y	in-silico (3D nasal cavity)	Dynamic
[205] Guo 2021	Endovascular	RRT	Catheter	Y	in-silico (cerebrovascular)	Static
Optimisation-based:						
[215] Lyons 2010	Transoral endotracheal	Mathematical model	Concentric tube robot	Y	in-silico (3D bronchus)	Static
[221] Gao 2015	Endovascular	ACO	Catheter	Y	in-silico (3D lower limb arteries)	Static
[218] Qi 2019	Endovascular	Mathematical model	Continuum robot	Y	in-vitro (blood vessels)	Static
[222] Li 2021	Endovascular	GA	Catheter	Y	in-silico (3D aorta and coronaries)	Static

Continued on next page

Table 4.2 – *Continued from previous page*

Ref.	Procedure	Method	Robot	Kinematics	Validation	Environment
[219] Guo 2021	Endovascular	Mathematical model	Catheter	Y	in-silico, in-vitro (3D vessel model)	Static
[220] Abah 2021	Endovascular	Mathematical model	Catheter	Y	in-vitro (3D cerebrovascular)	Static
Learning-based:						
[228] Zhao 2022	Endovascular	LfD using GAN	Guidewire	Y	in-vitro (3D vessel model)	Static
[229] Meng 2021	Endovascular	RL	Catheter	Y	in-silico (3D aorta)	Static
Level 2: Navigation using waypoints						
Learning-based:						
[230] Trovato 2010	Transanal	RL	Fibre optic endoscope	N	ex-vivo (3D swine colon)	Dynamic
[224] Rafii-Tari 2013	Endovascular	LfD using GMM	Catheter	N	in-vitro (3D aorta)	Static
[223] Rafii-Tari 2014	Endovascular	LfD using H+HMM	Catheter	Y	in-vitro (3D aorta)	Static
[13] Chi 2018a	Endovascular	LfD using DMPs	Catheter	Y	in-vitro (3D aorta)	Dynamic
[225] Chi 2018b	Endovascular	LfD using GMMs	Catheter	N	in-vitro (3D aorta)	Static
[227] Chi 2020	Endovascular	LfD using GAIL	Catheter	N	in-vitro (3D aorta)	Static
Level 3: Semi-autonomous navigation						
Node-based:						
[173] Qian 2019	Endovascular	Dijkstra	Guidewire	N	in-vitro (3D femoral arteries, aorta)	Static
[180] Girerd 2020	Transnasal	Potential field	Concentric tube robot	Y	in-silico (3D nasal cavity), in-vitro (origami tunnel)	Static
[178] Martin 2020	Transanal	Potential field	Endoscope	N	in-vivo (3D colon)	Dynamic
[179] Zhang 2020	Transanal	Potential field	Endoscope	Y	in-vitro (2D colon model)	Dynamic
[175] Cho 2021	Endovascular	Dijkstra	Guidewire	Y	in-vitro (2D vessel)	Static
[167] Fischer 2022	Endovascular	BFS	Catheter	Y	in-vitro (2D vessel)	Static
[174] Schegg 2022	Endovascular	Dijkstra	Guidewire	Y	in-silico (3D coronary arteries)	Static
Learning-based:						
[85] You 2019	Endovascular	RL	Catheter	N	in-vitro (3D heart)	Static
[231] Behr 2019	Endovascular	RL	Catheter	N	in-vitro (2D vessel)	Static
[232] Karstensen 2020	Endovascular	RL	Catheter	N	in-vitro (2D vessel)	Static
[233] Kweon 2021	Endovascular	RL	Guidewire	Y	in-vitro (2D coronary artery)	Static
[234] Pore 2022	Transanal	RL	Endoscope	Y	in-silico (3D colon)	Dynamic
[235] Karstensen 2022	Endovascular	RL	Guidewire	N	ex-vivo (2D venous system)	Dynamic

Reinforcement Learning: Trovato [230] developed a hardware system for a robotic endoscope that showed how classic RL algorithms such as State-Action-Reward-State-Action (SARSA) and Q-learning could be used to control the voltage for propulsion and determine the forward and backward motion of the robot.

However, the state-of-the-art in RL algorithms has shifted towards DRL which employs DNN to learn from high-dimensional and unstructured state inputs with minimal feature engineering to accomplish tasks [112]. Behr *et al.* [231], Karstensen *et al.* [232], and Meng *et al.* [229] proposed a closed-loop control systems based on DRL that use the kinematic coordinates of the guidewire tip and manipulator as input to generate continuous actions for each degree of freedom for rotation and translation. Karstensen *et al.* [235] showed the translation of this approach to ex-vivo veins of a porcine liver. The authors considered two control settings: a discrete action space and a continuous action setting. They found that DQN trains faster, while DDPG achieves more stable results and requires less domain-specific knowledge for reward calculation.

To further improve closed-loop control, You *et al.* [85] and Kweon *et al.* [233] automated control of the catheter using DRL based on image inputs in addition to the kinematic information of the catheter. They trained a policy in a simulator and showed its translation to a real robotic system using the tip position from an aurora sensor sent to the simulator to realize the virtual image input.

In addition to endoscopy, DRL is also being explored in other medical applications such as tracheotomy. Athiniotis *et al.* [12] used a snake-like clinical robot to navigate down the airway autonomously. They employed a DQN based navigation policy that utilizes images from a monocular camera mounted on its tip, which serves as an assistive device for medical personnel to perform endoscopic intubation with minimal human intervention.

4.3 Evaluation environments

The development of autonomous systems presents significant challenges during the design and validation phases. Implementing autonomous control algorithms on robotic systems without prior testing can lead to hardware failure and unpredictable behaviors, resulting in dangerous clinical situations and injuries [236]. Therefore, it is essential to have a controlled environment to evaluate algorithms without the risk of hardware breakdown.

Virtual environments provide an excellent platform to simulate both robotic and clinical scenarios [237]. Simulations are a safe, fast, and cost-effective solution that allows exploration of how autonomous robots should be designed and controlled for safe operation and optimal performance. Two primary ways in which simulations can aid in automating robotic tasks are discussed herein.

Firstly, simulations can be customized to model multiple agents, environmental conditions, and their interactions, thereby allowing for the analysis of the system's response to various settings, identifying potential problems and predicting hazardous situations. Simulations can be employed during the development stage to optimize system behavior and design fail-safe strategies. Additionally, during task execution, simulations can be utilized whenever decision-making is involved to predict the outcome of possible actions. Secondly, simulation provides a versatile environment for generating large amounts of data that can be used to train several MP algorithms. Successful learning requires a large database, which can be compensated for using realistic synthetic data generated by simulation

Additionally, such virtual setups can be used to train clinicians and surgeons in complex surgical procedures. In the following sections, we highlight the various simulated environments used in the context of IP and their deployment in realistic phantoms.

4.3.1 Simulation environment

There are several techniques used to govern the behavior of the instrument in different environments. A widely adopted method is the Finite Element Method (FEM) [238, 239], which involves dividing the tool into basic elements connected by nodes. The goal is to obtain a function that solves the equilibrium equations for the elements, incorporating geometry and material information [4, 10, 240]. In these studies, the instrument is modeled as discrete rigid bodies serially linked to one another. Another frequently used approach is the mass-spring model, which views the instrument as a network of masses connected by springs [6, 241]. In this method, the springs introduce flexibility into the model while constraining the distance between masses. Additional modeling methods include rigid multibody links that divide the instrument into a collection of rigid bodies joined by massless springs [242, 243], and hybrid methods that merge multiple techniques to model various segments of the instrument [244, 245].

The reconstruction of the luminal model involves the segmentation and rendering of medical images using software such as 3D Slicer [246] or VTK/VMTK toolkit library [247]. Previous studies have utilized CT images to perform simulations in a reconstructed 3D static bronchus for the guidance of bronchoscopy [248], or in a 3D static aorta model [169, 161, 213]. Similarly, the spatial anatomy of the nasal cavity was constructed based on the patient's CT medical image sequence [181, 183, 184]. Furthermore, Kuntz *et al.* generated real patient point cloud scenarios from endoscopic video of a patient's upper airway near the epiglottis and the colon [214].

The realistic simulation environment design also encompasses the instrument-lumen interaction, including contact forces and friction. The lumen is commonly considered a rigid object with a circular cross-section in most studies [241]. Collision detection in recent works is achieved by bounding volumes approximation [4, 2]. Regarding frictional forces, a quasi-static approach is often used, where velocities and accelerations are low, and the frictional effect is disregarded [4, 2]. In [7], the fluid dynamics of blood flow inside a vessel is investigated. The fluid dynamics is implemented and simulated on SimVascular [249], which is an open-source pipeline for cardiovascular simulation.

Recently, Behr *et al.* developed a simulated environment for cardiovascular IP using the SOFA library [231]. SOFA is an open-source library specially designed for interactive medical simulation and is considered as a benchmark for simulating medical scenes with accurate physics and rendering [250]. It allows objects to be represented with various bio-mechanical properties and visual displays, making it well-suited for developing complex, stable, and high-performance scenes. Meanwhile, You *et al.* utilized the Unity engine to simulate catheter movements, which allowed for body translation, rotation, and tip bending for three DoFs control [85]. The Unity game engine is known for its modularity and advanced features implemented in separate plugins, which makes it well-suited for medical simulation.

Athiniotis *et al.* developed a simulation for endo-tracheal intubation using the Gazebo simulator [185]. To simplify the task, the authors employed a follow-the-leader mechanism and limited the learning to the movement of the robot tip. Gazebo is a widely used robotic simulator that offers realistic rendering of environments and has been successfully applied in endoscopic robotic surgery [251].

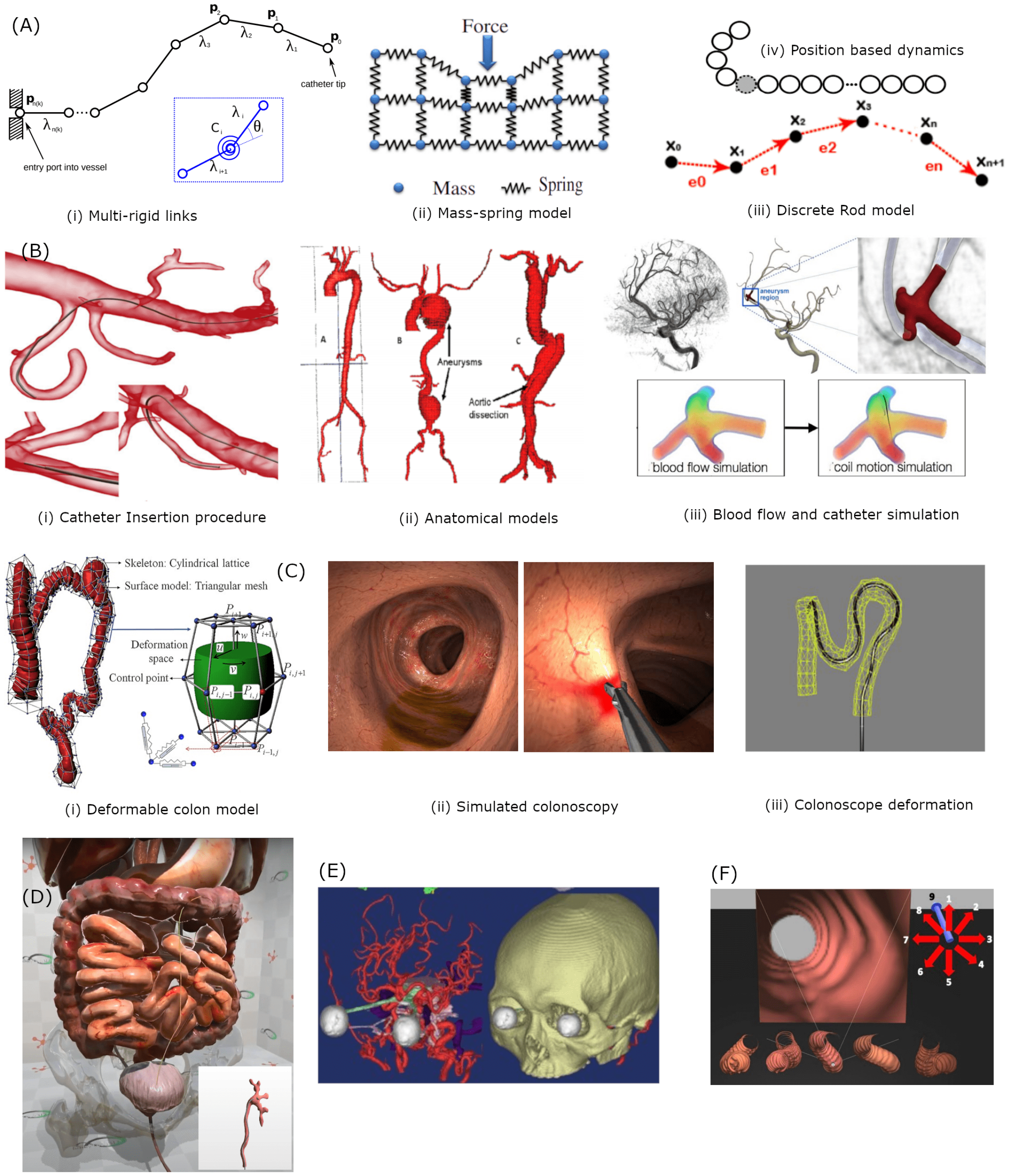
In the context of colonoscopy simulation, previous works have proposed various approaches. For instance, Yi *et al.* [252] developed a simulator with a haptic interface that allows for the "jiggling motion" to straighten the colonoscope and shorten the bowel. Meanwhile, De *et al.* [9] incorporated loop formation by modeling the tissues surrounding the colon. Jung *et al.* [8] presented a skeleton-driven real-time deformation model for the colon and endoscope, which consisted of a cylindrical lattice enclosing the triangle mesh of the colon surface.

Recently, PBD has been proposed by Muller *et al.* [253] and used for modeling continuum robots such as an urethroscope [5]. This technique is fast, stable and controllable and directly manipulates the particles of the mesh in a quasi-static manner, without the use of forces and impulses.

For transnasal surgeries, a 3D model of the brain and underlying structures was reconstructed using CT scans and MRI [11].

Table 4.3 Summary of publications on simulation environment for endoluminal navigation

Method	**References**	**Highlights**	**Physics engine/ graphics renderer**	**Lumen**
Point	[12]	Bronchoscope	Gazebo	Static
FEM (Rod-based)	[4], [10], [240], [231]	Catheters	OpenGL-Visual C++ SOFA	Static/Dynamic
	[254]	Colonoscopy	Ansys LS-DYNA	Dynamic
Mass-spring	[6, 241]	Catheters: Contact forces, friction	Visual C++, H3D, VTK	Small deformations
	[8]	Colonoscopy	Visual C++, OpenGL	Dynamic: Mass-spring
Hybrid	[245, 244]	Catheters: Collisions	Bullet, OpenSceneGraph	Static
Rigid-links	[85], [242, 243]	Catheters	Unity	Static
	[9]	Colonoscopy	OpenGL, Visual C++	Dynamic: Mass-spring
Position-based Dynamics	[5]	Urethroscope	CHAI3D (C++, OpenGL)	Static



(i) Multi-rigid links

(ii) Mass-spring model

(iii) Discrete Rod model

(i) Catheter Insertion procedure

(ii) Anatomical models

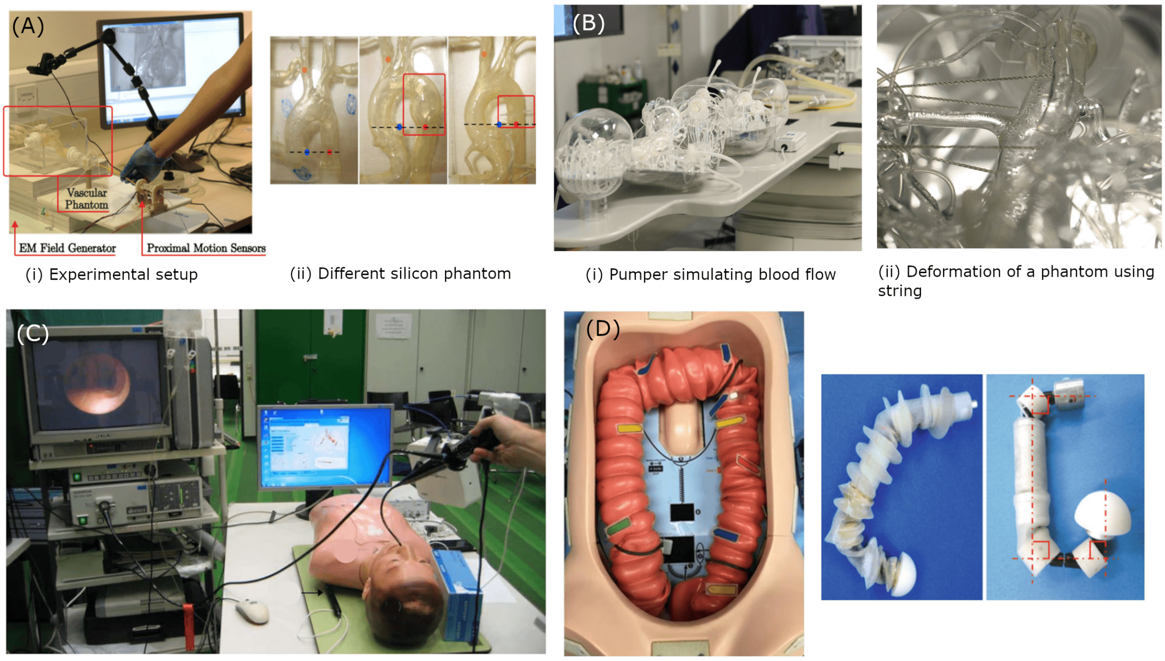
(iii) Blood flow and catheter simulation

(i) Deformable colon model

(ii) Simulated colonoscopy

(iii) Colonoscope deformation

Fig. 4.5 Schematics of simulation scenes used for endoluminal procedures. (A) Different models used for soft-object simulation [2–5], (B) Different simulation anatomies used for catheter insertion procedures, with additional constraints such as Blood flow, catheter motion and reconstruction from patient-specific data [4, 6, 7](C) Colon and endoscope deformation model for simulated colonoscopy [8, 9]. Dynamic cardiovascular lumen: Image superposition [10] (D) Urethroscopy procedure simulation with the catheter in SOFA [5] (E) 3D models for transnasal surgeries using MRI data and CT scans [11] (F) Simplified Bronchoscopy environment designed in Gazebo [12].



(i) Experimental setup

(ii) Different silicon phantom

(i) Pumper simulating blood flow

(ii) Deformation of a phantom using string

Fig. 4.6 Experimental setups for testing motion control in intraluminal procedures (A) Proposed experimental setups with different aortic arches [13] (B) blood circulation with a pumper and phantom deformations with a string [14] (C) Bronchoscopy setup for testing the guidance system [15](D) Various phantoms for colonoscopy [16]

4.3.2 Real Robotic setup

One crucial step in demonstrating the successful performance of a MP algorithm is to validate its feasibility in a realistic setup or phantoms. For instance, Schafer *et al.*[168] validated their method in a 3D rigid carotid artery phantom, while Wang *et al.* [160] carried out their experiments in a 3D rigid resin vessel phantom for catheterization. In their work, qi *et al.* and You *et al.* used a 3D printed heart model for their experiments [218, 85]. Zheng *et al.* reconstructed an abdominal aortic aneurysm using 3D pre-operative CT images and 2D intra-operative fluoroscopic images [14]. Simulation, phantom as in Fig. 4.6 (B) and patient data sets have been used to validate the proposed framework.

Rafii-Tari *et al.* [224, 223] and Chi *et al.* [13, 225] evaluated their proposed framework for cannulation of the left subclavian and right common carotid arteries using two silicone-based, anthropomorphic phantoms of the aortic arch. The robotic catheter driver, equipped with two servo motors and a PID controller, is used to drive the catheter along the desired trajectory. Navigation is facilitated by a camera placed on top of the phantom that provides a 2D projected image, and a graphical user interface that displays the current and upcoming positions. The catheter tip motion is captured using six-DoFs EM position sensors attached to the catheter tip.

Ratnayaka *et al.* [255] presented successful real-time MRI needle access of target vessels and endograft delivery in animal models, as shown in Fig.4.6(C). Fagogenis *et al.* [200] conducted in-vivo and ex-vitro experiments using a designed catheter in the blood-filled heart. Trovato *et al.* [230] conducted in-vivo and in-vitro experiments in swine colon using a standard robotic endoscope.

4.4 Limitations of MP and improvements

The need for automation in IP will increasingly demand the adoption of novel MP techniques capable of working in unstructured and dynamic luminal environments. MP for continuum robots is a complex problem because many configurations exist with multiple internal DoFs that have to be coordinated to achieve the desired motion [256, 38]. 32 of 65 publications consider MP for the robot without considering its kinematics, as shown in Table 4.2. This oversight suggests that future studies need to focus on the robotic constraints for active MP. Furthermore, replanning is required to adapt the current plan to deformable environments using sensorial information. The objective of replanning is to reduce the navigation error measured according to defined metrics. Therefore, the computational efficiency of MP becomes essential for real-time scenarios. This section highlights the limitations of MP that hinder their universal application in IP procedures and insights that can improve them.

Node-based: The searching strategy of node-based algorithms is based on specific cost functions. The optimality and completeness of the solution obtained using this strategy could be guaranteed. However, (i) node-based algorithms usually lack the consideration to satisfy robot capability during MP, such as robots' kinematic constraints; (ii) the uncertainty of sensing is rarely considered; (iii) the proposed methods are only applied in rigid environments, tissue deformations during procedures are not incorporated; (iv) node-based algorithms usually rely on the thorough anatomical graph structures. Accurate reconstructions of the anatomical environment in the preoperative phase are needed to build the data structure and search inside it. The mentioned limitations reduce the usability of these methods. In theory, they may work, but in practice, they are difficult to apply for autonomous real-time navigation in real-life conditions.

Some novel studies on the path planning of a steerable needle for neurosurgery could give some inspiration for IP, as these studies consider curvature constraints of a robotic needle. Parallel path exploration is used in the Adaptive Fractal Trees (AFT) proposed for a programmable bevel-tip steerable needle [257]. This method uses fractal theory and Graphics Processing Units (GPUs) architecture to parallelize the planning process, and enhance the computation performance and online replanning, as demonstrated with simulated 3D liver needle insertions. An Adaptive Hermite Fractal Tree (AHFT) is later proposed, where the AFT is combined with optimized geometric Hermite curves that allow performing a path planning strategy satisfying the heading and targeting curvature constraints [258]. Although developed and tested only for a preoperative neurosurgical scenario, AHFT is well-suited for GPU parallelization for rapid replanning.

Sampling and Optimization based: Sampling and optimization-based techniques are capable of considering robot-specific characteristics, but their efficacy is significantly influenced by the robot's model. For soft continuum robots, such as those used in IP, the modeling approaches and the incorporation of soft constraints for obstacle avoidance present significant challenges that remain under investigation [24]. While sampling-based techniques have the advantage of reducing computational time as compared to optimization-based approaches, they do not necessarily guarantee optimality of the solution, owing to their intrinsic property of random sampling. Therefore, finding a feasible path solution is not always guaranteed with these methods. On the other hand, optimization-based approaches are often time-consuming and applied mainly in static environments for preoperative MP.

Hybrid approaches, which combine different methods, have the potential to improve performance and address the limitations of individual methods. Learning-based approaches, which are emerging in the field, can be integrated with other methods to overcome their limitations. For instance, Wang *et al.* proposed a hybrid approach for MP in narrow passages by combining RL and RRT algorithms [259]. This approach improves local space exploration and ensures efficient global path planning. Other authors have also proposed hybrid MP methods for IP navigation. For example, Meng *et al.* presented a hybrid method using BFS and GA for micro-robot navigation in blood vessels of rat liver, with the aim of minimizing energy consumption [156].

Research in optimization-based methods is also ongoing, particularly for achieving optimal preoperative planning under complex constraints. For example, Granna *et al.* implemented Particle Swarm Optimization (PSO) for a CTR system in neurosurgery [260], while Pourmanda *et al.* employed dynamic programming for micro-robot path planning in rigid arteries based on a minimum effort criterion [261]. However, achieving intraoperative MP requires techniques for reducing the search space of constrained optimization problems. In this regard, Howell *et al.* propose an augmented Lagrangian trajectory optimizer solver that can handle general nonlinear state and input constraints while offering fast convergence and numerical robustness [262]. In the context of IP MP, an optimization solver with reduced search space could be potentially applied for efficient intraoperative planning.

Learning-based:

The recent shift towards learning-based approaches, as shown in Fig. 4.4, has proven successful in adapting to unseen scenarios. Therefore, the interest of this thesis lies in the application of RL for IP subtask automation.

DRL requires a huge amount of training data due to their inherent complexity, a large number of parameters involved and the learning optimization [110]. Therefore, a massive amount of data need to be acquired, moved, stored, annotated and queried in an efficient way [263]. In the surgical domain, high-quality diverse information is rarely available [75]. Various groups have proposed shared standards for device integration, data acquisition systems and scalable infrastructure for data transmission such as the CONDOR (Connected Optimized Network and Data in Operating Rooms) project (https://condor-h2020.eu/) and OR black box [264]. A general trend to overcome data limitation is through the use of simulators. Therefore, the first milestone of this thesis is to develop a realistic simulator described in Chapter 5 and Chapter 8.

One of the major concerns with implementing DRL is safety [265]. DRL relies on DNN, which may exhibit unexpected behavior for unseen data beyond the training regime. The guarantee of safe behavior using DNN is still an open problem, necessitating the incorporation of safety constraints to avoid hazardous actions. Some studies have proposed safe-RL frameworks for safety-critical applications that use barrier functions to limit robot actuation within a secure workspace. To address this issue, a safe-RL framework that employs reward shaping and Formal Verification FV tools has been proposed and is discussed in Chapter 6 and Chapter 9.

The performance of commonly used DRL methods is highly sensitive to the hyper-parameters settings, and may vary substantially between runs. It is unlikely that a single RL algorithm would perform equivalently in a heterogeneous robotic control problem. Even for closely related tasks, appropriate methods need to be carefully selected. The user must determine when there is sufficient prior knowledge, and when learning can begin. Reliable and safe learning is a challenge that can be broadly classified into two groups: (1) reducing sensitivity to hyper-parameters, and (2) reducing issues associated with local optima.

To address the former challenge, we experiment with state-of-the-art algorithms that are robust to hyper-parameter settings, such as PPO, and with methods that can automatically tune their own hyper-parameters, such as Soft Actor Critic (SAC). The second challenge to reliable and stable learning is local optima, which can arise due to unstable policy updates. To overcome the problem of unstable policy updates, when the step size between successive policy updates is too large, we use f-divergence methods, such as KL-divergence, to constrain the policy search from being greedy [266].

LfD is a commonly used approach for learning human-like gestures [53]. However, its drawback lies in the requirement for a large number of demonstrations for proper training, which is often impractical in clinical settings due to time, resource and ethical constraints. Additionally, LfD only allows the robot to perform as well as the human demonstrations, as significant deviations from the demonstrated behavior can result in unstable policy learning [266]. To address this issue, we propose implementing divergence minimization between the expert and the learning policy [267] through the use of GAIL. GAIL allows the use of demonstrations to guide the exploration during the learning phase, reducing the time required to find an improved control policy that departs from the demonstrated behavior[144]. Moreover, it facilitates the convergence towards a policy that performs better than the demonstrations provided. The combination of imitation learning and RL losses ensures that the policy eventually outperforms the demonstrated behavior and avoids significant deviation during training.

An example of LfD-based planning method for multi-section continuum robots is presented by Seleem *et al.*, who propose two novel approaches to generate motion demonstrations: a flexible input interface that allows humans to demonstrate different motions for the robot end-effector and the Microsoft Kinect sensor, which provides motion demonstrations faster via human arm movements [268]. Future prospects include designing a user-friendly human-machine interface to collect useful demonstrations and developing methods to combine exploration-based RL with the collected demonstrations.

Other drawbacks of robotic DRL and directions that can be taken to mitigate them are elaborated in Chapter 9.

4.5 Conclusions

In summary, MP is a critical aspect of IP automation, and the development of efficient and reliable MP algorithms is essential for the adoption of autonomy in the clinical setting. In this chapter, we provide a comprehensive overview of MP techniques used in IP. We categorize the MP methods into four types: node-based, sampling-based, optimization-based, and learning-based. We highlight the limitations associated with these methods and provide suggestions for future research. Node-based methods are simple and easy to implement but may not work well for complex environments. Sampling-based methods are more suitable for complex environments but require more computational resources. Optimization-based methods aim to find the optimal solution but can be computationally expensive. Learning-based methods require large amounts of data but are more efficient and can adapt to changes in the environment.

Further research is needed to address the limitations of existing methods and to develop new methods that can overcome the challenges associated with automation in clinical settings.

Contributions of this chapter

1. Taxonomy of MP methods for IP
2. Limitations and improvement in MP for effective IP navigation.

Publications linked to this chapter

1. Ameya Pore, Zhen Li, Diego Dall'Alba, Albert Hernansanz, Elena De Momi, Arianna Menciassi, Alicia Casals, Jenny Denkelman, Paolo Fiorini and Emmanuel Vander Poorten."Autonomous Navigation for Robot-assisted Intraluminal and Endovascular Procedures: A Systematic Review", accepted in Transactions on Robotics (T-RO).

2. Di Wu, Renchi Zhang, Ameya Pore, Diego Dall'Alba, Xuan Thao Ha, Zhen Li, Yao Zhang, Fernando Gonzalez, Elena De Momi, Wojtek Kowalczyk, Alícia Casals, Jenny Dankelman, Jens Kober, Arianna Menciassi, Paolo Fiorini. "A Review on Machine Learning in Flexible Surgical and Interventional Robots: Where We Are and Where We Are Going", submitted to IEEE Transactions on Automation Science and Engineering

Chapter 5

Colonoscopy Navigation using deep visuomotor control

5.1 Introduction

In 2020, there were 1.9 million new cases of CRC detected globally, resulting in a mortality of 935 thousand people [269]. The World Health Organization (WHO) predicts an average annual increase of 3% worldwide for the next two decades [270]. Early detection is crucial for improving the survival rate, which decreases to below 5% at Stage IV and is close to 100% at Stage 0 [271]. Colonoscopy screening programs are considered the most effective method for detecting and treating lower-gastrointestinal pathologies, particularly CRC, which is the third most prevalent form of cancer globally [272]. The screening process involves the insertion of a FE up to the rectum. Then, the FE is slowly withdrawn while searching for early-stage CRC lesions. However, as discussed in Chapter 2, FE-based procedures are complex and require extensive training to master due to the non-intuitive mapping between the endoscope tip and the control steering knobs [83].

As a result, these procedures are susceptible to human errors, which causes significant discomfort and pain to patients due to the tissue stretching associated with FE manipulation [187]. One of the main causes of pain is looping, where the FE advances into the colon without a corresponding progression of the tip. Looping also increases the risk of colon perforation and massive bleeding [28]. Additionally, endoscopists are at risk of work-related musculoskeletal injuries due to awkward neck and body posturing [273]. Furthermore, the shortage of adequately trained endoscopists compared to the increasing clinical demand for colonoscopy procedures can result in the potential loss of human lives [274].

To address the limitations of traditional FEs, researchers have been investigating various methods to improve the performance of colonoscopy. One promising approach is the use of robotic systems, which have been shown to provide improved dexterity, precision, and control over traditional manual colonoscopy. Wireless capsule endoscopes have been developed since

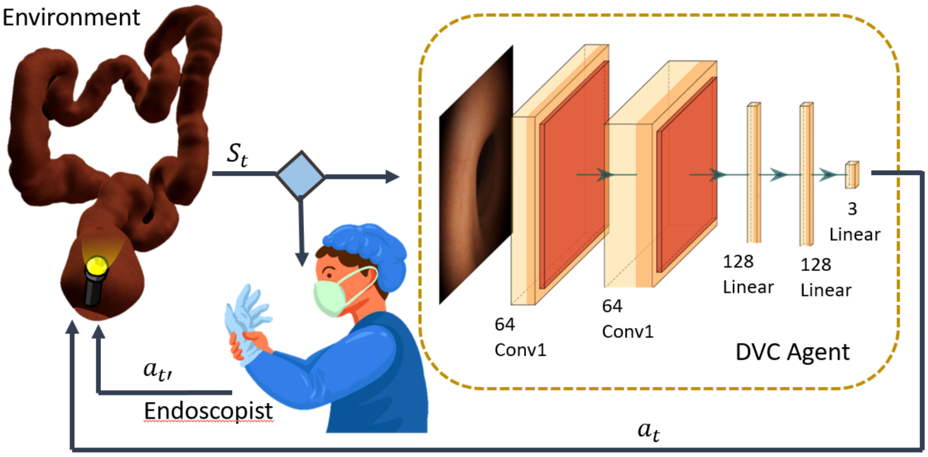


Fig. 5.1 Deep Visuomotor Control (DVC) flow diagram. The environment provides a state observation S_t. The DVC agent uses the state input to generate an action a_t that is applied to the environment. During the training phase, DVC learns a task-conditioned policy π_ϕ to perform autonomous colonoscopy navigation. In the evaluation phase, the clinicians can supervise and override DVC decisions through action $a_{t'}$.

they are non-invasive, painless and do not require sedation [275]. However, these devices lack the control of the endoscopic point of view, increasing the risk of missing pathological areas [83]. Therefore, current research efforts are focused on developing navigation systems using robotized FE, such as the STRAS system [276], or magnetic actuated FEs [277].

Robotized FEs introduce automation technologies to enhance human operator abilities, particularly by adding autonomous navigation, which is the most time-consuming step of a routine colonoscopy procedure [178]. By allowing endoscopists to focus on the clinical aspect of the procedure rather than the manual control of FE, the overall procedure outcome can be improved, and training time reduced [187].

During the navigation phase, the clinician primarily uses visual feedback from the FE camera to advance through the lumen [278]. A common gesture observed during a colonoscopy procedure is to centralize the target direction of the endoscope towards the lumen center. Rule-based controllers have been developed to replicate this gesture by reducing the distance error between the image center and the detected lumen center [54]. However, these algorithms fail when the tip of the endoscope approaches close to the colon wall due to loss of lumen's center view and camera occlusion. This frequently occurs due to the colon's highly deformable nature and variable mobility caused by patient movements, peristalsis, and breathing, leading to changes in lumen diameter and haustral folds that make lumen detection challenging. Such situations require human interventions to correct the motion direction, or they can be handled by adaptive exploration methods, as proposed in this work.

Rule-based controllers are gradually being replaced by data-driven approaches such as DRL since they provide greater degree of adaptability [51, 52]. However, the use of DRL in learning surgical task policies has been limited to low-dimensional physical state features such as robot kinematic data, which are considered to be sample-efficient and easy to learn [52, 279]. This thesis proposes an image-based DRL approach for endoscopic control (Fig. 5.1) that focuses on learning the navigation task by developing an end-to-end [1] policy to map raw endoscopic images to the control signal of the endoscope, called Deep Visuomotor Control (DVC). We evaluate the DVC control primarily through a user study with 20 expert GastroIntestinal (GI) endoscopists who perform the navigation task in a realistic virtual simulator.

While the introduction of autonomous navigation can improve clinical practice by relieving clinicians from demanding cognitive and physical tasks, maintaining human supervision is highly desirable in safety-critical areas such as medical robotics to address ethical and legal concerns [66]. Therefore, it is necessary to consider human-in-the-loop for the deployment of DVC in realistic surgical scenarios. Thus, we conducted a second user study with 20 novice participants to demonstrate that non-expert users can easily supervise autonomous navigation and that DVC reduces the need for human intervention compared to a state-of-the-art method.

5.2 Automated control for colonoscopy

Magnetic guided endoscopes have been the subject of several studies [277, 178, 187]. However, extending the navigation methods presented in [277] to complex non-linear trajectories is challenging, as it is based on following simple predefined trajectories. While heuristic path planning algorithms are used to generate a feasible path in a colon map [187], this approach employs force-based real-time sensing for navigation, which is still not widely available in existing endoscopic devices. Furthermore, interpreting robotic actions without scene visualization is challenging and not suitable for human supervision.

In [178], a static perception model is developed, which extracts the center of the lumen from raw image observation, with control of the endoscope position and orientation imparted by a proportional controller that aligns the endoscopic image with the center of the lumen. However, similar rule-based controllers previously developed in [54] require significant manual tasking for non-linear components such as analytically computing image jacobian and interaction matrix [280]. Moreover, lumen detection could be unstable and prone to errors due to the dynamic nature of the colon and its sharp bends. These scenarios require a vision-based control system to improve during policy training, which is limited with hand-engineered features for perception [280].

[1] An end-to-end mapping refers to a type of model architecture that directly maps the input data to the output data, without using intermediate representations or feature engineering. In this context, it refers to a single neural network that related the raw endoscopic image date to the corresponding control signal of the endoscope without any intermediate steps

Learning end-to-end visuomotor representations for direct control using DRL overcomes these limitations without separately designing perception and control models and offers the ability to improve model parameters while training [45, 281].

Several frameworks have been proposed in the literature for training DRL policies to automate surgical tasks involving manipulation of rigid and deformable objects [51, 52, 282, 283]. These frameworks utilize simplified environments specifically designed for robot-assisted surgery to learn instrument control during the procedure. Recently, [284] proposed a DRL method for optimizing the endoscopic camera viewpoint. However, the low-dimensional state information used for training DRL algorithms, such as kinematic values of the robot, position of target etc. [51, 52, 282, 284], may not be sufficient to capture the complexity of real colonoscopy scenarios where accurate endoscope kinematics cannot be captured due to sensing limitations [187], and intra-operative guidance is solely based on visual feedback.

5.3 Simulation platform

As discussed in Chapter 4, simulations provide a safe and cost-effective solution for testing and validating the behavior of the system in various environments, identifying potential issues and designing fail-safe strategies. Despite the numerous advantages of simulation, there remain several challenges that hinder its extensive use in robotics, such as the difficulty in selecting and calibrating models. Defining a scenario for robotic simulation requires the choice of various models, including robot dynamics, perception systems, environment, and interactions with the environment. These models require the selection of a large set of parameters, which can be daunting as the complexity of models increases, such as in highly deformable and dynamic environments with friction between objects and multiple interaction agents. Parameterizing models can be a time-consuming and tedious process, relying on ad-hoc parameter identification strategies or trial-and-error attempts to fine-tune the model until the desired behavior is achieved. Model selection also involves a trade-off between accuracy and computation time, depending on the application. Selecting a complex model without correct parameters may lead to worse results that take longer to obtain than simpler models.

To address these challenges, we build a colonoscopy simulation framework [2] using the popular graphics engine and a real-time 3D development platform Unity with the integration of SOFA and Unity Machine Learning Agents toolkit (ML-Agents). SOFA allows to create complex medical simulations such as organ deformation and collisions, while ML-Agents enables Unity to serve as an interactive platform for training of neural network based agents using DRL.

[2] This development was carried out as a collaborative effort with Early-Stage Researcher (ESR) 15 of the ATLAS project, described in Chapter 1. ESR 15 further used it for developing a hands-free user interface with several input devices such as the haptic and the joystick devices.

5.3.1 Colon simulation

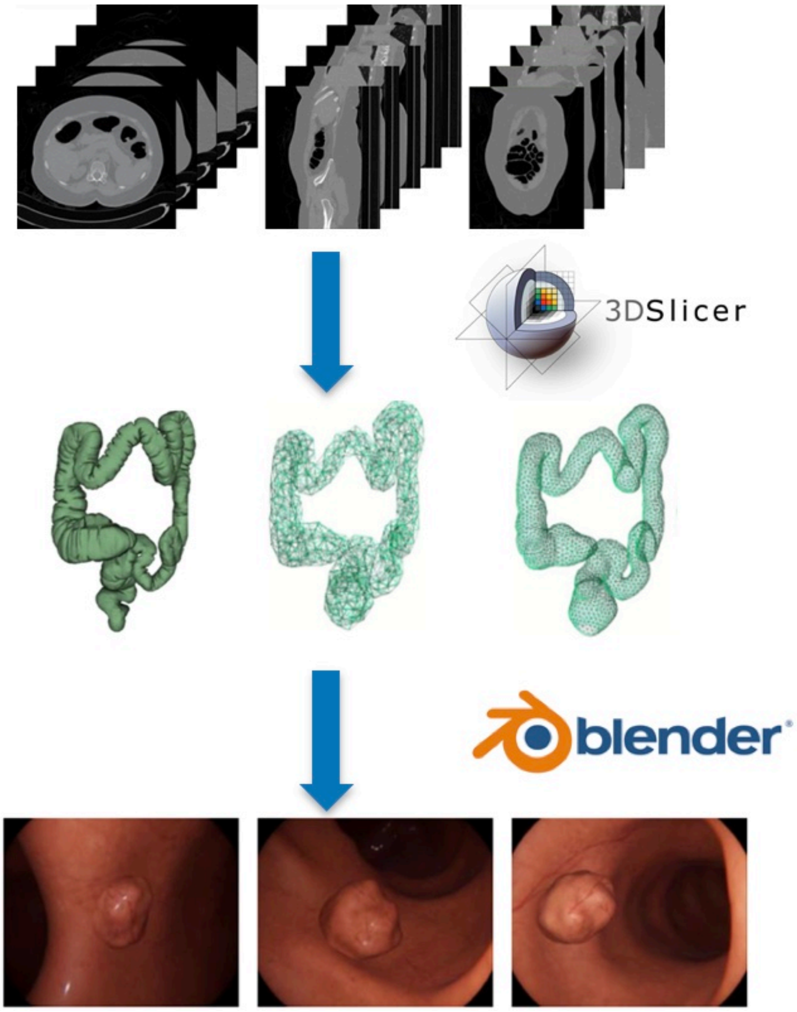


Fig. 5.2 3D colon model construction. The 3D model is extracted from real patient CT scan to generate volumetric meshes. Mucosa textures from real endoscopy images are added to the meshes [17].

In order to create realistic colon simulation, a CT colonography dataset from the Cancer Imaging Archive is utilized to derive the colon models [285]. A semi-automated segmentation approach is used to segment the 3D models of the bowel [286], as depicted in Fig. 5.2. The segmented models are further refined, and volumetric and superficial meshes are generated by importing them to Blender. As a subsequent step, textures are generated using the Kvasir dataset, which comprises actual endoscopy images obtained from different patients [287]. To create the primary mucosa texture, a combination of various endoscopy images is stitched together and applied to the inner surface of the model. This process ensures the generation of clear, continuous, and non-blurry mucosa walls. Following the creation of the main mucosa textures, veins are incorporated onto the walls by extracting vein networks from images within the Kvasir dataset. To extract the veins from real endoscopic images, a grayscale conversion is performed, and median filtering is applied to the pixels. Furthermore, a contrast-limited adaptive histogram equalization technique is employed to enhance the contrast of the grayscale image, making the veins appear darker than their surroundings. These extracted veins are then applied to the mucosa texture images, utilizing random distributions to determine the location, rotation, and size of the veins, with mean and standard deviation values determined empirically. Finally, the 3D model is unwrapped, and the resulting mesh model is divided into rectangular segments. These segments are uniformly projected onto the created UV texture map, ensuring repetitive placement throughout the model.

To enhance the visual quality and realism of the default Unity pipeline, we have incorporated the High Definition Render Pipeline (HDRP). The HDRP focuses on differentiating materials under various lighting conditions while ensuring consistent illumination, thereby ensuring that all objects in the scene interact with light in a uniform manner. The HDRP shaders offer several features that contribute to achieving more realistic visuals and simulating real endoscopy images. For example, the addition of a white coat mask to the organ material creates a reflection effect on the 3D organ surfaces when illuminated by light sources. Additionally, HDRP allows us to mimic the characteristics of endoscopic camera views, such as vignetting, fish-eye distortion, and chromatic aberration. Vignetting refers to the darkening of the periphery in an endoscopy image, while chromatic aberration manifests as blurred edges between areas of high contrast. These effects are directly applied to the image buffer of the virtual camera, enabling real-time rendering capabilities.

In order to simulate the soft deformable mechanics of organs, we have integrated SofaAPAPI-Unity3D, an interface that enables Unity's PhysX Engine to leverage SOFA's more physically accurate models for tissue deformation [17]. This approach ensures that the colon simulation is not only accurate in terms of mechanical behavior but also visually realistic, which is crucial for training and education purposes.

Endoscope simulation - The simulation scenario is similar to a magnetically guided FE where external magnets control the motion of the magnetic tip while the tether follows the tip passively. However, we neglect the effect of the endoscope tether in this preliminary simulator version due to multiple collision points with the colon model that could lead to simulation instability. The endoscope tip is modeled as a rigid capsule with a weight of 20g, a length of 36mm, and a diameter of 14mm. An angular drag of 4 rad/sec^2 is added to account for the frictional resistance. The endoscope tip has four degrees of freedom for motion, including translation (insertion/retraction), roll, and bending in two perpendicular directions (pitch/yaw), as shown in Fig. 5.3. The endoscope tip also embeds a camera, which allows for visual inspection during the simulation process.

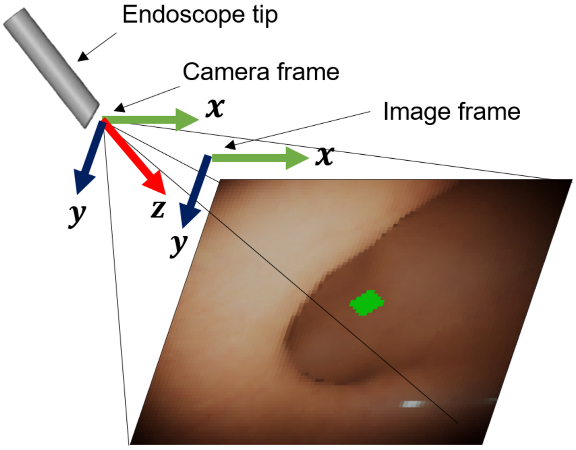


Fig. 5.3 Representation of the local frame at the endoscope tip. The X-Y plane of the camera is parallel to the image frame, while the z-axis represents the direction of insertion. Tip bending is carried out on the X-Y plane while the roll is carried on the z-axis. DVC uses a low-resolution image as state input. The green region represents the detected lumen center.

5.4 Deep Visuomotor control

DRL background - The colon navigation problem is formalized into a MDP represented by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \gamma, T)$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ is the action space, $\mathcal{P}$ is the transition probability distribution, $\mathcal{R}$ is the reward space, $\gamma \in [0,1]$ is the discount factor and T is the time horizon per episode. At each time step t, the environment produces an observation $s_t \in \mathcal{S}$. The agent generates an action $a_t \in \mathcal{A}$ based on a policy $\pi(s_t)$, applies it to the environment, and receives a reward $r_t \in \mathcal{R}$ [129]. As a result, the agent transitions to a new state s_{t+1} drawn from the transition function $p(s_{t+1}|s_t, a_t)$, $p \in \mathcal{P}$, or terminates the episode at state s_T.

Learning algorithm - The agent's goal is to learn a stochastic behavior policy π with parameters ϕ, $\pi_\phi : \mathcal{S} \rightarrow \mathcal{P}(\mathcal{A})$, that maximizes the expected future discounted reward $E[\sum_{i=0}^{T-1} \gamma^i r_i]$. We selected PPO [129] as the DRL algorithm for our study, as it has demonstrated high performance in terms of wall-clock training time and hyper-parameter tuning. Although our aim was not to propose a novel DRL method, but rather to conduct a user study to evaluate the performance of image-based DRL in colonoscopy navigation. The PPO algorithm comprises a value and policy network that use shared parameters to predict the action vector (a) and estimate the state value (V), respectively. In the training session, we set the length of each episode as 10k iteration steps, $\gamma = 0.99$, and the batch size and the learning rate hyperparameters as 64 and 3e-4, respectively. The PPO clip ratio was 0.2, with 4 mini-batches per epoch and 4 epochs per iteration. We added a loss term proportional to negative policy entropy, with a coefficient of 0.01. Each training session lasted for 1.5 million iteration steps, which was sufficient for the reward function to converge (Fig. 5.7).

Action space - During the preliminary manual control of the endoscope, it was observed that the visibility of the lumen was hampered when the endoscope was directed towards the colon wall, especially at sharp turns. Hence, it is crucial to avoid endoscope translation in such scenarios. Therefore, we devised an action strategy where translation motion with a constant velocity of $v_{end} =$ 10mm/sec is executed only when the lumen is detected. The action space is composed of discrete angular rotation values in the three degrees of freedom at the endoscope tip, $\delta\theta_j = \alpha$, $\alpha \in 0, -1, +1$ in the j^{th} spatial dimension. In the tip local reference frame, $j \in x, y, z$ corresponds to the alignment of orientation in the horizontal and vertical directions in the image plane and the endoscope roll, respectively (Fig. 5.3). In cases where the lumen is not visible, the endoscope's translation velocity is set to zero, allowing only orientation changes to detect the lumen.

Observation space and policy - The sensory input to the DVC agent is composed of down-scaled endoscopic images. The RGB images rendered by the endoscopic camera (1024x1024 pixels) are downscaled to 128x128 pixels. This down-scaling was carried out to reduce the computational complexity of the training process (i.e. sample efficiency and wall-clock training time), based on prior RL literature [39, 134]. The policy π_ϕ is represented by a CNN architecture consisting of two convolutional layers (as depicted in Fig. 5.1) that encode visual scene representations. The details of the network are publicly available on the project website[3]. The output of the CNN is fed into a combination of fully connected layers and a LSTM layer to represent time-dependent behavior. Each layer has 128 rectified units, followed by linear connections to the output logits π_t for each action a_t and a value estimate V_t. A softmax function transforms the logits into action probabilities. The entire network is trained end-to-end to acquire task-specific visual features.

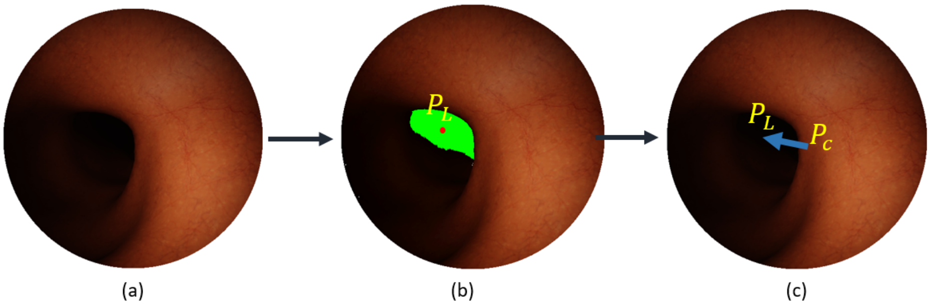


Fig. 5.4 Proposed adaptive threshold segmentation pipeline for lumen detection. Each RGB frame captured by the endoscopic camera is passed through the adaptive filter to detect the dark pixels a) original RGB frame b) Image mask for the detected lumen (in green) c) distance vector between the image center P_c and the centroid of the detected darkest regions P_L.

[3] https://github.com/Ameyapores/DVC

Table 5.1 Navigation parameters used for validation with their description

	Navigation metrics	Description
1	Time of insertion (TOI)	TOI is measured from the time point where the initial movement of the endoscope is detected to the time point when the caecum is reached.
2	Perforation	Perforation refers to the scenario when excessive force is applied on the colon wall (especially at the turning point) that can lead to severe injuries. Studies based on tensile property analysis of human rectal tissue reported the maximum elongation of 62% [288]. The average diameter of the colon models used is 5cm, hence a threshold of $\delta d = 3cm$ is decided to classify the deformation as perforation.
3	Normalized distance traveled	Distance traveled is crucial as multiple backward motions, reversing the direction, can lead to suboptimal trajectories. The distance traveled is measured using the position values of the endoscope tip. This distance is normalized by the centerline distance of the colon model in order to compare among different colon models. Normalized distance above 1 indicates a path distance longer than the centerline path, while a normalized distance below 1 indicates a shorter path than the centerline was followed.
4	Average LD	Lumen centralization is believed to create smooth insertion trajectories hence the lumen distance in the image plane is recorded at each timepoint. This distance is normalized by the size of the image to get a value in [0,1]. Lumen distance value 0 denotes that the image center (P_c) coincides with the detected lumen (P_L), and value 1 denotes that the detected lumen is at the farthest point.

Reward function - The objective of the navigation task is to complete the procedure without any complications by successfully tracking the colon. The successful tracking of the colon is achieved when the lumen center P_L is close to the image center P_c. To this end, we design a dense reward function $r_t(s_t, a_t)$ as follows:

$$r_t(s_t, a_t) = \begin{cases} C(1 - (||P_L - P_c||_2 / D_{max})), & L = 1 \\ -1, & L = 0 \end{cases} \tag{5.1}$$

where $D_{max} = 1/2 * (Imagewidth) = 64$, is the normalization factor which is the maximum distance possible, L represents the lumen detection flag, (1 denotes lumen detected, 0 denotes no lumen detected), the hyperparameter C is chosen as 1. Additionally, the agent is provided with a reward of +10 upon reaching the end of the colon and -10 if it returns to the starting point, in order to incentivize unidirectional movement towards the caecum.

In order to detect the colon lumen in the endoscope image, a real-time threshold segmentation algorithm is implemented [289]. The algorithm is capable of running at 30 frames per second. Initially, the image is segmented to detect the darkest and most distinct region, with the assumption that this region contains the distal lumen with the highest probability. To perform the segmentation, the RGB image is converted to grayscale, and a circular region is cropped from the center of the image with a diameter equal to the image width, to eliminate the vignette effect on the corners. The resulting segmentation is illustrated in Fig. 5.4.

5.5 Experimental Validation

The experimental goal is to compare the navigation performance of the DVC agents, the baseline rule-based control method [178], and the endoscopists. To achieve this objective, a pipeline is created to record the position and orientation values of the endoscope, lumen distance in the image space, colon deformations, and camera image in the developed simulator. These parameters are synchronized and recorded during the experiments using the *labstreaminglayer* software, which is a unified system for collecting time-series measurements [290].

5.5.1 Endoscopist data acquisition

A group of 20 expert GI endoscopists (with more than four years of experience) were asked to make navigation attempts in the colonoscopy simulation scene developed in Sec. 5.3. Due to the time constraints and COVID regulations at the hospital[4], only four colon models could be selected considering the opinion of domain experts to represent progressively more complex scenarios (Fig. 5.5). The endoscopists were instructed to navigate the colon models from the rectum to the caecum using a PlayStation joystick device. The colon model C_0, which depicts a simplified colon model that conforms with the shape and size of the average human colon, was used to familiarize the endoscopists with the controls before beginning the trials. The trials began with the endoscopist's attempts on the C_1 colon, followed by randomized attempts on C_2 and C_3. The randomization between C_2 and C_3 was introduced to identify performance bias based on the colon model.

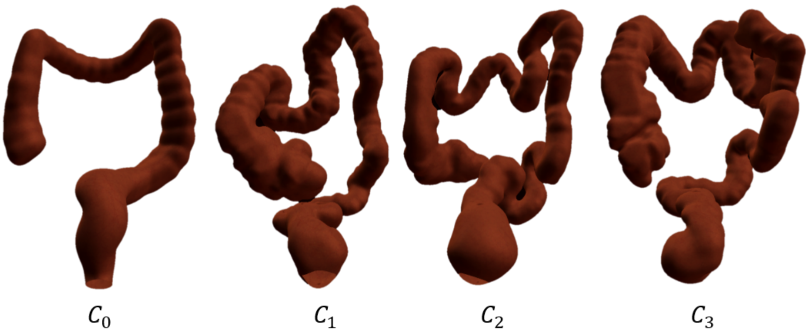


Fig. 5.5 Colon models used in the experimental phase. (From left to right) ranked in increasing complexity order, C_0, C_1, C_2 and C_3 colon models. The model complexity is characterized by the centerline distance of the model from rectum to caecum, and the number of acute bending, i.e. >90 degree, which is estimated through visual inspection.

[4] Ospedale Le Molinette (Torino, Italy)

5.5.2 Training DVC

We conduct three experiments to validate DVC's performance in navigation. Firstly, we aim to determine the sample efficiency of training DVC on different levels of colon complexity. We train DVC agents separately using the same colon models as those used in the endoscopist experiment.

Secondly, we establish a comparative analysis between DVC and endoscopists by following a similar experimental workflow as in the endoscopist experiments. In this experiment, DVC is only trained on the C_0 model and tested on C_1, C_2, and C_3 colons.

Thirdly, we train DVC on the C_0 model, followed by training on the C_1 model to test if training on a complex colon after a simple one improves performance. To ensure that the overall iteration steps for DVC training are limited to at 1.5 million, we terminate training on C_0 after 1 million iteration steps and then load it back to train on C_1 for 500k iteration steps. Table 5.3 provides an overview of the experiments.

5.5.3 Supervision

To evaluate the performance of the rule-based controller and the DVC in a supervision task, we recruited 20 novice participants with no endoscopy experience. The participants were asked to supervise the navigation of the endoscope through C_1, C_2, and C_3 colon models, using one of the following control strategies in each trial:

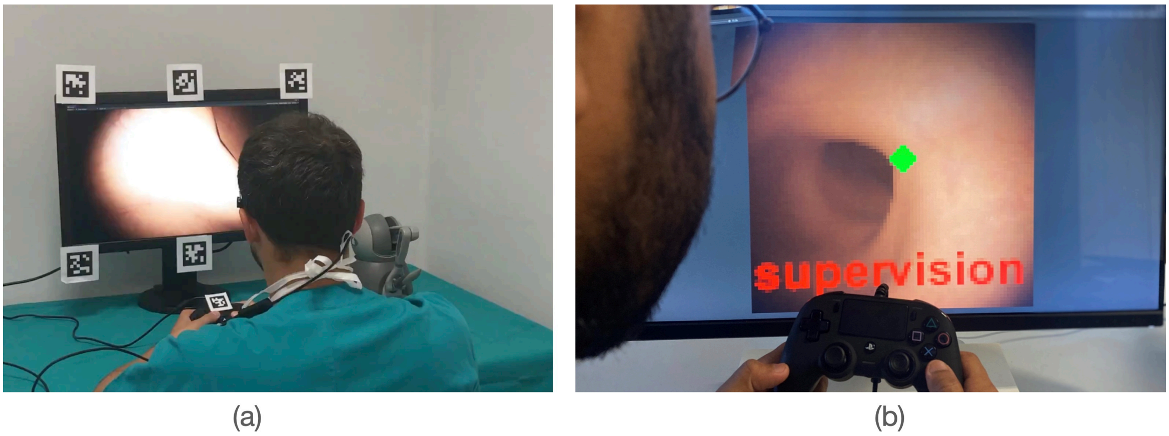


Fig. 5.6 (a) Navigation experiments with endoscopists (b) Novice user supervision while navigating by autonomous control strategies. *supervision* is printed on the screen, indicating the switch to manual control. When the endoscope is oriented towards the lumen (green point), the user can give back the control to the autonomous agent. A low-resolution (128x128 pixels) image is displayed to facilitate interpretability of machine decisions, however users have the option to change to high resolution (1024x1024 pixel) display.

1. Manual control: Participants were instructed to control the endoscope exclusively using a joystick throughout the procedure.

2. Rule-based baseline [178]: A proportional controller was generated for orientation control that aligned the image center (P_c) to the detected lumen (P_L) using the Lumen Distance (Lumen Distance (LD)) as follows:

$$\delta\theta = \beta \begin{bmatrix} P_{L_x} - P_{c_x} \\ P_{L_y} - P_{c_y} \end{bmatrix} \tag{5.2}$$

   The rule-based controller required manual supervision when the lumen center was not detected (Fig. 5.6).

3. DVC: A fully trained DVC_{C_0} was deployed. The DVC was given 50 iteration steps ($\Delta_t = 50$) to search for the lumen when it was not detected. After Δ_t steps, the DVC notified the requirement of human supervision, and manual control was activated. The user had an override option to take control when unsafe behavior was encountered, e.g., collision with the colon wall or reversing the direction of motion. Once manual control was active, participants could navigate the endoscope safely and return control to the DVC or rule-based controller.

The number of interventions by the participant was recorded during each attempt. After all the trials, participants completed a NASA Task Load Index (TLX) questionnaire [291] to score their human-perceived workload.

Data Analysis - The navigation performance is evaluated using four different parameters. Time of Insertion (TOI) and the number of colon perforations are qualitative assessment measures for colonoscopy procedures [292]. In addition, average LD and the normalized distance traveled are two metrics introduced in this study to measure the accuracy of the trajectories. The details of each parameter are elaborated in Table 5.1. Any navigation attempt where the user or DVC reversed direction of motion and returned to the rectum, or caused heavy perforation to destabilise the colon model, is considered a failed attempt.

5.6 Results and discussion

The learning curves for the DVC trained on different levels of colon complexity are shown in Fig. 5.7. The colon model C_0 is the simplest, and therefore the DVC agent achieves high reward values in relatively fewer steps than in other colon models. A high reward indicates successful completion of the navigation task. In contrast, the C_2 model is highly complex, and the agent requires 1.2 million steps to achieve high-reward convergence. The learning curve for the C_1 model lies between those for C_0 and C_2, indicating that training time is related to colon complexity. However, it is important to note that DVC_{C_0} is capable of navigating other complex colon models, indicating that it has acquired task-specific features that can be generalized to other colon models (Table 5.3).

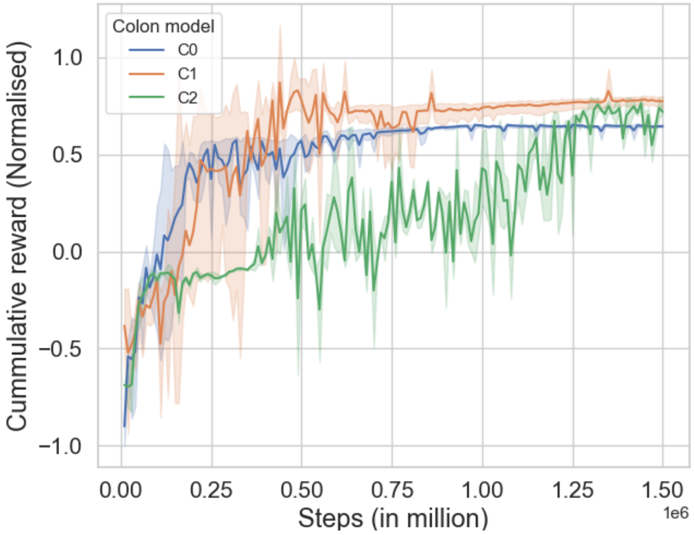
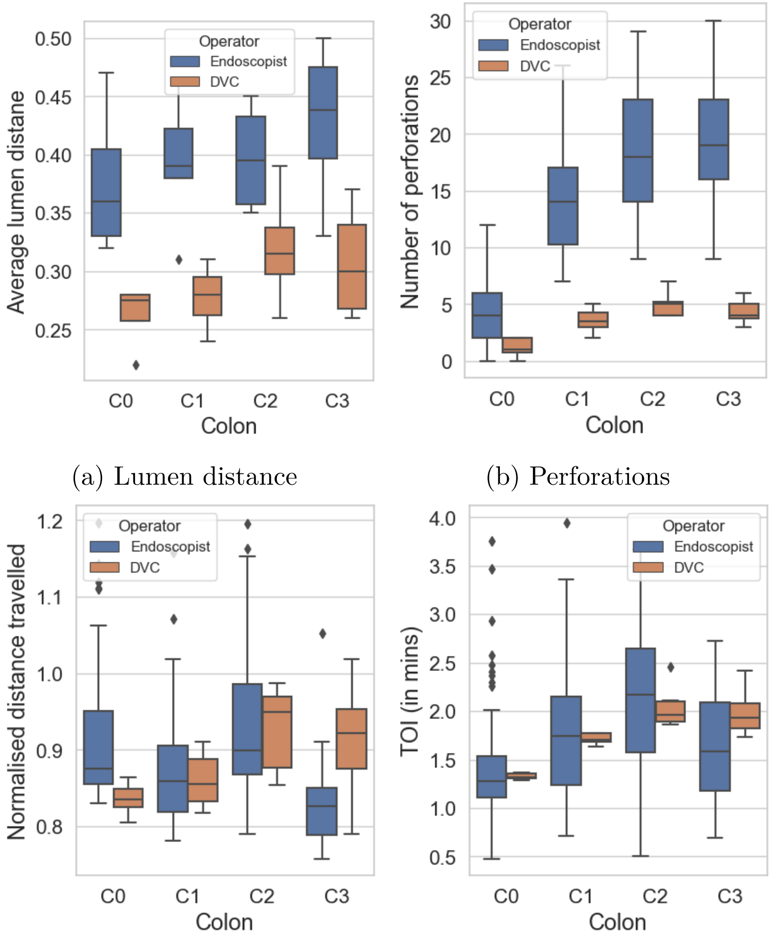


Fig. 5.7 Learning curve of DVC trained on varying complexity of colon, using three colon models. Cumulative reward is normalized in the range $[-1,1]$. The shaded area spans the range of values obtained when training the agent starting from five different initialization seeds.

Comparative analysis - The results of a comparative analysis between the performance of 20 endoscopists and 10 different DVC agents trained on the C_0 model are presented. The simulation was validated by expert clinicians, who positively evaluated the joystick used to navigate the endoscope as intuitive, user-friendly and easy to learn. The comparison of average LD, number of perforations, completion time and normalized distance traveled showed significant differences between the endoscopists and DVC. DVC demonstrated precise tip centralization and less number of perforations compared to endoscopists. This difference may be attributed to clinicians' tendency to push the colon wall at acute bends due to the rigid constraints of clinically available FEs. On the other hand, DVC is trained on reward feedback to minimize LD and stay centralized to avoid contact with the wall. The normalized distance and TOI did not show substantial differences between the two groups, but there was more variance in the endoscopists' performance. Some followed convoluted trajectories, while others followed smoother trajectories, resulting in higher or lower normalized distances and TOI. Trajectories executed by endoscopists and DVC agents for different colon models are shown in Fig. 5.9, where the smoothness of a trajectory is estimated using a jerk index J (cm/sec^3). The average performance of DVC agents remained consistent across different colon models, while the endoscopists showed wide variance in their optimal trajectory performance.

The results of splitting the training process into two colon models, $DVC_{C_0+C_1}$, and evaluating its performance on other colon models are presented in Table 5.3. Notably, $DVC_{C_0+C_1}$ shows improved lumen detection performance compared to DVC_{C_0}, which reaches high rewards at 500k iteration steps, indicating no additional feedback to improve the



(a) Lumen distance

(b) Perforations

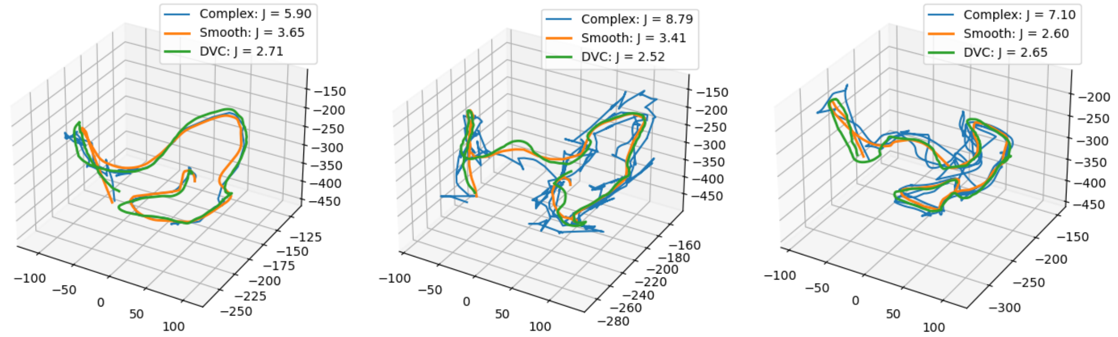
(c) Normalized distance

(d) Time of insertion

Fig. 5.8 Navigation performance comparison plots between DVC and endoscopists. Several parameters are plotted a) Lumen distance, b) perforations, c) Normalized distance, d) Time of insertion.

Table 5.2 NASA Task Load Index for novice users. Lower score indicate good user experience

	Manual control	Rule-based control	DVC
Mental demand	63	33	**18**
Physical demand	65	38	**9**
Temporal demand	30	47	**17**
Performance	25	34	**12**
Effort	57	38	**10**
Frustration	42	41	**12**
Mean workload	47	38	**13**



(a) C_1 (b) C_2 (c) C_3

Fig. 5.9 Trajectory plot of DVC, complex and smoothest endoscopist performance for a) C_1 b) C_2 3) C_3 models respectively.

Table 5.3 Comparison between DVC and Endoscopists

	DVC_{C_0}				$DVC_{C_0+C_1}$			
	Average LD	Perforation	TOI	Normalised distance	Average LD	Perforation	TOI	Normalised distance
C_0	0.27±0.01	0.5±0.25	1.37±0.05	0.84±0.02	0.24±0.02	1±1	1.32±0.03	0.84±0.02
C_1	0.30±0.01	3.3±1.5	1.74±0.04	0.88±0.08	0.25±0.01	3.3±0.5	1.70±0.07	0.82±0.03
C_2	0.36±0.03	5±1	2.22±0.2	0.97±0.03	0.28±0.01	4.6±0.5	1.89±0.03	0.85±0.01
C_3	0.35±0.02	4.6±0.5	2.15±0.23	0.92±0.08	0.29±0.03	3±1	1.78±0.05	0.89±0.09
Mean	0.31±0.04	5.0±1.2	2.20±0.75	0.90±0.04	**0.23±0.04**	**4.3±1.2**	**1.96±0.59**	**0.86±0.04**

performance. We speculate that DVC_{C_0} may reach suboptimal local minima. In contrast, when DVC_{C_0} is loaded to train on C_1, it encounters acute bends that offer the potential to maximize the cumulative reward. However, there is no significant improvement observed in other navigation parameters, including perforation,TOI, and normalized distance.

Supervision - The study included two types of human interventions: those where the user overrides the control due to unsafe behavior, and those where the system demands human supervision. The rule-based baseline required an average of 5 ± 1.8 human interventions for user override and 2.5 ± 1.5 interventions when the system demanded human control. In contrast, the DVC system required an average of 0.1 ± 0.5 human interventions for user override and 0.05 ± 0.2 interventions when the system demanded human control. This difference can be attributed to DVC's adaptability in searching for new insertion directions when the lumen is not easily detected, a feature lacking in the rule-based controller.

The study also evaluated the participants' perceived workload using the NASA-TLX survey. The results showed that manual control and rule-based controller were more demanding in all task load categories compared to DVC. Participants reported a substantial workload reduction when using DVC. Table 5.2 presents the NASA-TLX scores for each control strategy.

5.7 Conclusion

Autonomous colonoscopy navigation has been an area of active research, but prior works have relied on heuristic control policies that cannot adapt to situations where detecting lumen is challenging, leading to frequent human interventions. To overcome this limitation, we propose an end-to-end DRL-based controller (DVC) that learns a mapping between endoscopic images and the endoscope's control signal, such as tip orientation. The proposed method has been validated in a simulated environment for colonoscopy that closely mimics the soft tissue dynamics of the colon tissue.

The simulation platform is modular, scalable, and open-sourced, and it can receive inputs from different devices and systems, such as playstation joystick, keyboard, and haptic devices. Furthermore, it is realistic in terms of timing, visual, and mechanical rendering, combining CPU and GPU implementations. The navigation performance of DVC has been compared to the motion data acquired from 20 GI endoscopists. The experimental validation shows that DVC has an equivalent performance in terms of the time of insertion and the distance traveled. However, DVC reduces the number of perforations and shows efficient lumen tracking, improving safety. Moreover, a novice user study has been conducted to demonstrate that supervision of DVC control significantly reduces the user workload, with overall performance comparable to expert GI endoscopists.

While the results are promising, there are some limitations of this work. First, it is not straightforward to know the direction of motion of the endoscope. Hence, the newer version of the virtual scene will simulate the endoscope body dynamics, providing the insertion length. Second, if the robot needs to learn from raw image observations, it also needs to evaluate the reward function from raw image observations, which requires a hand-designed perception system. This can be mitigated by using online user interaction through human-in-the-loop RL, which can be implemented by using the eye gaze tracking collected during the acquisition of motion data from the GI endoscopists while guiding the navigation.

The results obtained in Table 5.3 present an opportunity to study curriculum learning-based setup, where colon navigation can be trained in increasing levels of colon complexity. Future work will demonstrate the formal validation of the realism of the proposed virtual simulator. Overall, the proposed method can significantly reduce the workload of the endoscopists and improve the safety of colonoscopy navigation. It has the potential to be deployed in real-world scenarios with further development and validation.

Contributions of this chapter

1. Autonomous colonoscopy navigation using DRL based approach that combines visual perception and motor control in an end-to-end manner, called DVC.

2. The approach is evaluated through a user study with novice and expert endoscopists, demonstrating the feasibility and potential of the proposed method. The study shows that the proposed approach is able to improve navigation accuracy, while reducing the workload and cognitive effort required by the endoscopist.

Publications linked to this chapter

1. Ameya Pore, Martina Finocchiaro, Diego Dall'Alba, Albert Hernansanz, Gastone Ciuti, Alberto Arezzo, Arianna Menciassi, Alicia Casals, and Paolo Fiorini.“Colonoscopy Navigation using End-to-End Deep Visuomotor Control: A User Study”, In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 9582-9588. IEEE, 2022.

Chapter 6

Constrained Reinforcement Learning for Safe Colon Navigation

6.1 Introduction

The development of autonomous navigation systems has been proposed as a promising approach to improve the performance of colonoscopy. Autonomous navigation systems based on visual information use different processing techniques, which rely on the assumption that the region of maximum depth within an image represents a valuable target for immediate heading adjustment [293]. The assessment of lumen depth using the contours of surrounding structures has been proposed in [26, 294]. However, large structures such as haustral folds may not be noticeable in images, specifically in the presence of obstruction or at sharp turns when an endoscope faces the colon wall for most of the movement. Rather than the contours, most methods for autonomous navigation rely on recognizing the darkest or deepest region in endoscopic images to adjust the endoscope heading. These techniques are typically simple to execute and many studies have examined different techniques for segmentation of dark regions utilizing optical flow [295] or image intensity [293, 178]. All of these approaches require hand-engineered visual features, which are generally arduous to create. In contrast, Lazo *et al.* proposed a CNN-based approach for lumen segmentation that does not require hand-crafted solutions and tends to generalize better when trained on a large amount of data [296].

Irrespective of the method used for detection, a rule-based controller is commonly used to minimize error with respect to the center of the endoscopic image. These controllers are typically based on the Proportional-Integral-Derivative approach [296, 178, 26] or finite state machines [294], but they are not robust to changes in the estimates provided. Such changes can arise from errors in the segmentation method or dynamic deformations of the anatomy. To address these limitations, an end-to-end mapping between endoscopic images and control signals was proposed using a DRL approach in Chapter 5. This approach requires a reward function that minimizes Lumen Distance (LD), which, in turn, requires lumen detection. In this work, we aim to develop an end-to-end DRL approach that is independent of a separate perception system.

Despite the promising results of applying DRL in robotic systems, the implementation of such methods in real-world surgical settings raises concerns about safety. Safety is a critical aspect of surgical procedures, and any system used in such settings must prioritize patient well-being. One major challenge of DRL methods is their vulnerability to unanticipated behaviors in situations that were not encountered during training. These unanticipated behaviors could result in potentially harmful consequences for patients [297, 298]. Therefore, it is crucial to thoroughly evaluate the safety of any robotic system that employs DRL methods before clinical deployment.

Constrained Reinforcement Learning (CRL) provides a way to tackle safety by restricting the agents from taking potentially unsafe actions through the incorporation of an additional cost function that should be minimized. While the reward function incentivizes desirable behavior, the cost function penalizes unwanted actions. However, achieving a perfect zero-cost result through numerical optimization in DRL is often impractical, so a threshold is set as a maximum acceptable value for the cumulative cost. Examples of algorithms to face this challenging problem include the Interior-point policy optimization (IPO) algorithm [299], which utilizes a logarithmic barrier function, Constrained Policy Optimization (CPO) [300], based on the concept of safe policy improvement or Safety-Oriented Search (SOS) [301], which incorporates a genetic step in the training loop. This thesis focuses on the L-PPO algorithm, which utilizes the Lagrangian dual relaxation of a constrained optimization problem [302]. The L-PPO algorithm offers a simple and efficient method for updating constraints, while inheriting all the strengths of the PPO algorithm, such as trust-region policy improvement and first-order optimization.

The use of safety specifications in CRL methods approximates the risk of a state over the trajectory, but does not guarantee overall safety [303]. It is therefore important to ensure that the agent never makes decisions that could result in safety violations. This validation requires estimating the risk of safety violations without executing the action. Running the network over many experiments and collecting the unsafe configurations can be time-consuming and can only give an empirical evaluation without any guarantee of safety [304]. To overcome these limitations, this thesis proposes the use of Formal Verification (FV) to mathematically guarantee that the agent's actions remain in the safe regime before deployment. However, the high-dimensional, non-linear, and complex structure of DNN used in DRL presents a challenging NP-Hard problem for providing formal guarantees [305].

Several investigations have proposed various methods to address the challenge of applying FV to real-world DNNs [304], including quantitative verification techniques [306]. Recently, researchers have extended the FV framework to DRL systems [307, 308] and applied it to robot-assisted MIS [309]. However, previous FV methods applied to robot-assisted surgical

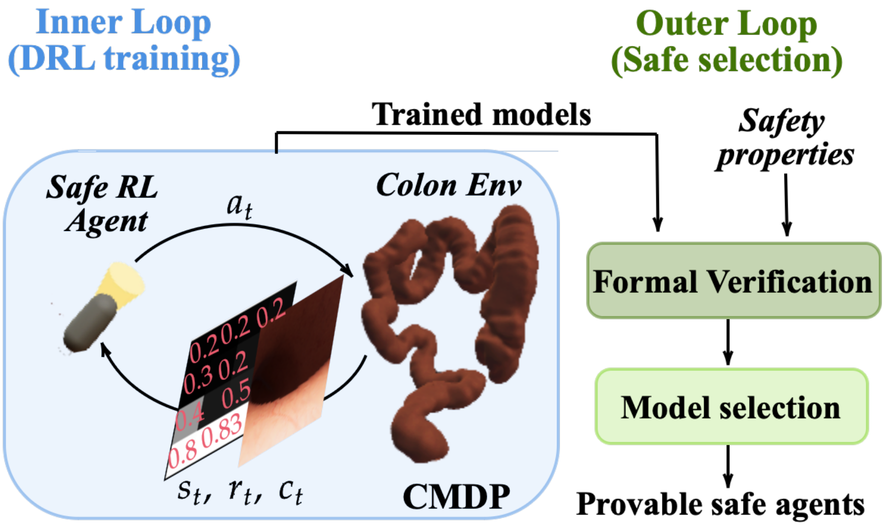


Fig. 6.1 Safe-RL framework proposed in this work. Agents are trained in a Constrained MDP setting with soft constraints. The trained policies are examined with the FV tool which identifies the safety violations. Policies without safety violations are selected for final deployment to ensure a completely safe behavior.

setups merely identify states that may result in safety violations, without utilizing the outcomes for other scalable goals, such as model selection. In this research, we employ VeriNet, an advanced FV tool [310], to create a model selection strategy that can formally verify a vast array of policies and detect those that do not exhibit any safety violations, ensuring complete safety (as shown in Fig. 6.1).

Thus, our proposed framework integrates the following characteristics: (1) An end-to-end DRL technique for colon navigation, which removes the necessity for a distinct lumen detection system. (2) A CRL methodology that restricts the policy in a predetermined safe state-space to decrease the likelihood of dangerous actions. (3) A model selection approach that chooses policies that meet all safety requirements, with each policy formally verified to detect safety violations.

The proposed framework is assessed in a virtually simulated colonoscopy environment that precisely imitates the colon tissue's dynamics. We evaluate the colon navigation performance and safety of the proposed CRL approach compared to the standard DRL approach. By utilizing the model selection strategy, we identify policies that guarantee complete safety for CRL while none for DRL. This study emphasizes that the integration of CRL and FV can enhance safety in autonomous colonoscopy navigation.

6.2 Problem Statement

In this section, we briefly introduce the safety objectives for autonomous navigation. The colonoscopy environment developed in Chapter 5 is used as a validation testbed.

6.2.1 Overview of the safety Framework

The traditional approach of relying on the region of greatest depth as the immediate heading adjustment goal is vulnerable to various external factors such as lighting conditions, focal length, and the surrounding tissue geometry, leading to an unreliable target for precise navigation [296, 178, 26]. Moreover, it overlooks the 3D structure of the surrounding anatomy, restricting the endoscope's ability to navigate through tight bends (as shown in Fig. 6.2) where many images may not be well-centered within the lumen (as depicted in Fig. 6.3a). Consequently, the endoscope may capture close-up views of the lumen wall that are highly illuminated due to reflection. Therefore, defining safety objectives that prevent the endoscope from moving orthogonally to the colon wall, which may cause perforation, can lead to a safer trajectory.

Henceforth, we establish two safety indices for our study: (1) Soft constraints that guide the agent to avoid collisions with the colon wall through safety probability analysis and optimization. Incorporating these constraints during training leads to CRL, where the agent learns to take actions that conform to safe configurations. (2) Hard constraints that impose strict restrictions on the system to prevent it from entering unsafe regions, such as perforation. We identify that movement towards the illuminated region of the image can result in a trajectory orthogonal to the wall. To address this, we introduce a set of four hard constraints, denoted as safety properties (Θ), based on brightness thresholds in different image regions. If the threshold is exceeded, the robot must restrict its actions in that direction.

Enforcing hard constraints is a challenging task for CRL methods as they often rely on indirect constraints based on the expectation of cumulative cost [304, 300]. In our work, we aim to leverage FV techniques to analyze the policy and evaluate its adherence to the specified hard constraints.

6.3 Constrained Reinforcement Learning

6.3.1 Deep Reinforcement Learning

The aim of a DRL algorithm is to determine a policy that maximizes the expected cumulative reward over a trajectory. Mathematically, this can be represented as follows:

$$\max_{\pi_\theta \in \Pi} J_r(\pi_\theta) := \mathbb{E}\tau \sim \pi\theta[R(\tau)] \tag{6.1}$$

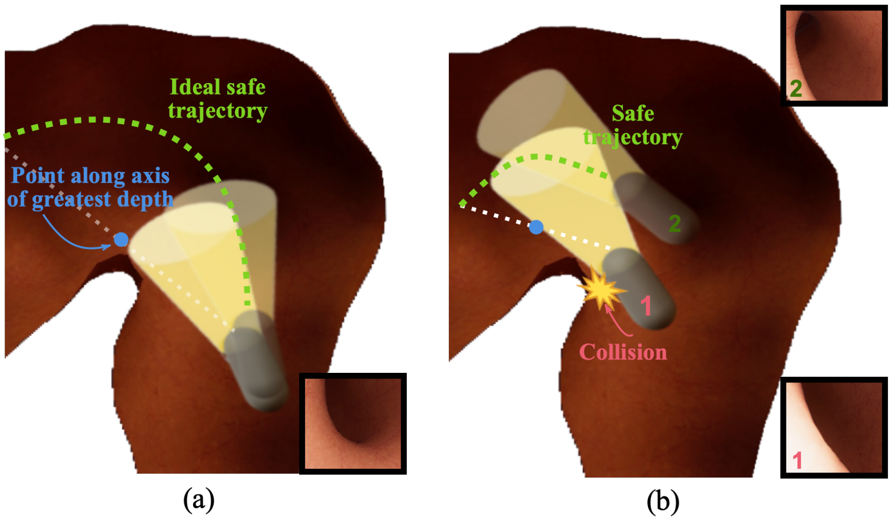


Fig. 6.2 An illustration of a capsule endoscope positioned inside the lumen facing an upcoming turn, showing the point of greatest depth (darkest point) and the lumen center. Navigation towards the deepest point can lead to biased motion towards the inner wall of the turn, potentially resulting in camera occlusion and collision with the wall, as demonstrated in (a) and (b). The right-hand side square boxes display the endoscopic view, with (a) depicting the similar views of two endoscope tips, while (b) shows that endoscope 1 approaches the wall more closely than endoscope 2 while following the line of greatest depth.

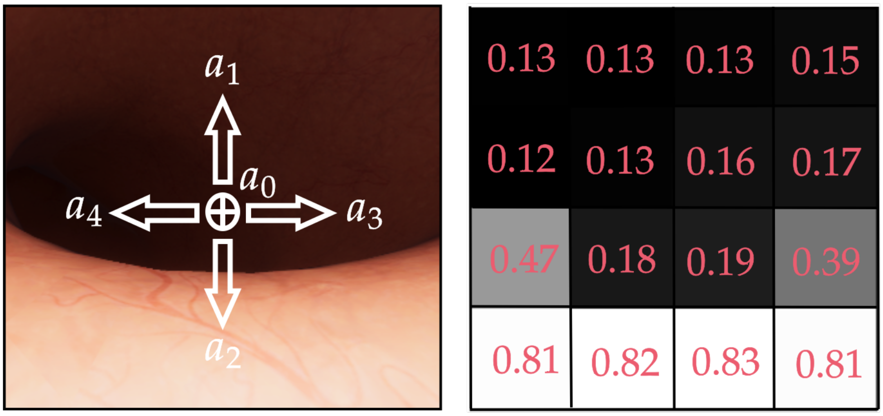
Here, π_θ represents a policy parameterized by θ, τ refers to a trajectory, and $R(\tau)$ denotes the cumulative reward received along that trajectory. Similar to Chapter 5, we continue to use PPO as a consolidated algorithm for incorporating the safety constraints.

Observation Space

The observation space is characterized as a low-dimensional discretization of the endoscopic image, represented by a 4x4 matrix, as shown in Fig. 6.3b. The image is first divided into four regions along each dimension, and each square region is assigned a normalized average value of the underlying pixels. This discretization step is carried out to simplify the definition of safety properties and preserve the local features of the image. The resulting 2-D down-scaled image is then flattened into a list of 16 values, which form the input to the DRL algorithm.

Action space

The action space consists of five distinct actions, which are associated with the movement in one of the four cardinal directions (a_1 :`up`, a_2 :`down`, a_3 :`right`, and a_4 :`left`), and an additional action a_0 :`center` that sets the angular velocity to zero. The agent moves at a constant linear velocity of 3 mm/s. The angular velocity, which determines the rotation of the endoscope tip, is dependent on the specific action selected and corresponds to a fixed



0.13	0.13	0.13	0.15
0.12	0.13	0.16	0.17
0.47	0.18	0.19	0.39
0.81	0.82	0.83	0.81

(a) (b)

Fig. 6.3 (a) Endoscopic view with the allowed actions. (b) Discrete representation of the input space used for the agent.

angle of 0.017 rad/s in the two degrees of freedom. To facilitate the input-output mapping, the DNN controller has been designed with 5 output neurons (one for each action) and 16 input nodes, which is consistent with the discretized image representation discussed in the previous subsection.

Reward function

:

We design a reward function that incentivizes the agent to reduce the distance from the end of the colon while minimizing the interactions with the colon wall. The function provides a high positive reward to the agent for successfully completing the task, a small penalty for touching the colon wall, and an additional penalty that scales with the distance between the agent and the end of the colon. The mathematical expression of the reward function is as follows:

$$R_t = \begin{cases} 10 & \text{reaches the end} \\ -\beta & \text{touches the wall} \\ (-dist_t) \cdot \eta & \text{otherwise} \end{cases} \tag{6.2}$$

where $dist_t$ is the centerline distance from the end of the colon at time t. The centerline distance for each colon model is estimated prior to training using checkpoints. η is a normalization factor, and β is a fixed penalty for each collision. The values of η and β are empirically set to 0.001 and 0.01, respectively, in our experiments.

6.3.2 Constrained DRL and Lagrangian-PPO

In the preceding sections, we provided an overview of the optimal policy for an MDP. However, in scenarios where safety is critical, it is crucial for an agent to ensure additional essential behaviors that supersede the achievement of the primary task [302]. For example, during colonoscopy navigation, preventing the perforation of the lumen wall is of greater importance than reaching the destination, despite the latter being the primary objective [294].

To address this issue, a Constrained Markov Decision Process (CMDP) is typically used to model the problem. CMDP is an extension of a standard MDP that incorporates an additional signal, the cost function, denoted as $C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and a threshold value $d \in \mathbb{R}$ that limits the expected value of the cost. Although we consider only one cost function and its corresponding threshold for simplicity, the framework can be extended to handle multiple constraints. The set of feasible policies for a CMDP is formally defined as:

$$\Pi_{\mathcal{C}} := \{\pi_\theta \in \Pi : \ \forall k, \ J_C(\pi_\theta) \leq d\} \tag{6.3}$$

where $J_C(\pi_\theta)$ is the expected cost function over the trajectory and d is the corresponding threshold.

To find a policy $\theta \in \Pi_{\mathcal{C}}$ that maximizes the reward while satisfying the constraints, a constrained DRL algorithm can encode this problem as a constrained optimization problem in the following form:

$$\max_{\pi_\theta} \quad J_r(\pi_\theta), \quad \text{s.t.} \qquad J_C(\pi_\theta) \leq d \tag{6.4}$$

One feasible approach to integrate the constraints in an optimization problem is by utilizing *Lagrange multipliers*. In the context of DRL, a possible technique is to transform the constrained problem into its dual unconstrained counterpart. The objective function for optimization can be expressed as follows, which can be maximized using any policy gradient algorithm:

$$J(\theta) = \min_{\pi_\theta} \max_{\lambda \geq 0} \mathcal{L}(\pi_\theta, \lambda) \tag{6.5}$$

where $\mathcal{L}(\pi_\theta, \lambda) = J_r(\pi_\theta) - \lambda(J_C(\pi_\theta) - d)$.

For maximizing the objective function, various DRL algorithms can be utilized, with one of the common choices being PPO. Recent studies have reported promising results by using Lagrangian dual optimization together with PPO [302]. In our proposed framework, we incorporate a cost function into the same reward function used in PPO. The cost function assumes a value of 1 only when the capsule interacts with the wall, and 0 otherwise. A cost threshold value of 500 is selected to balance the safety of the capsule while maintaining its reward performance, as discussed in Sec. 6.5.

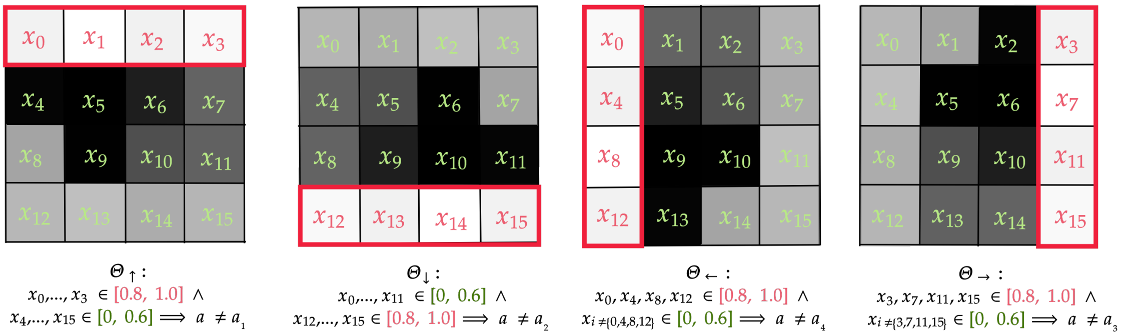

x_0 x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10} x_{11} x_{12} x_{13} x_{14} x_{15}
$\Theta_\uparrow$:
$x_0,\ldots,x_3 \in [0.8,\ 1.0] \land$
$x_4,\ldots,x_{15} \in [0,\ 0.6] \implies a \neq a_1$
$\Theta_\downarrow$:
$x_0,\ldots,x_{11} \in [0,\ 0.6] \land$
$x_{12},\ldots,x_{15} \in [0.8,\ 1.0] \implies a \neq a_2$
$\Theta_\leftarrow$:
$x_0, x_4, x_8, x_{12} \in [0.8,\ 1.0] \land$
$x_{i \neq \{0,4,8,12\}} \in [0,\ 0.6] \implies a \neq a_4$
$\Theta_\rightarrow$:
$x_3, x_7, x_{11}, x_{15} \in [0.8,\ 1.0] \land$
$x_{i \neq \{3,7,11,15\}} \in [0,\ 0.6] \implies a \neq a_3$

Fig. 6.4 Illustration of four safety properties designed, namely $\Theta_\downarrow, \Theta_\uparrow, \Theta_\rightarrow$, and $\Theta_\leftarrow$. When the scope is close to the upper, lower, left, or right lumen wall, the respective row/column squares in the input space have high illumination with values in $[0.8, 1]$, hence the agent should not move in that direction.

6.4 Formal Verification

The FV of DNNs can be represented by the mathematical tuple $\mathcal{R} = \langle \mathcal{F}, \mathcal{P}, \mathcal{Q} \rangle$ [304, 305]. Here, $\mathcal{F}$ denotes a trained DNN, $\mathcal{P}$ represents a precondition for the input, and $\mathcal{Q}$ defines a postcondition on the output. The precondition $\mathcal{P}$ specifies the allowable input configurations, while the postcondition $\mathcal{Q}$ describes the required output results that must be verified.

The verification process entails demonstrating the existence of at least one concrete input vector $\vec{x}$ that satisfies the given constraints, as expressed in the following assertion:

$$\exists ; \vec{x} ; | ; \mathcal{P}(\vec{x}) \land \mathcal{Q}(\mathcal{F}(\vec{x})) \tag{6.6}$$

The verification algorithm utilizes a search procedure to determine if an input vector $\vec{x}$ that satisfies the precondition $\mathcal{P}$ and the postcondition $\mathcal{Q}$ exists, and outputs `SAT` if such an input exists [305].

In this study, we introduce a set of safety properties, represented as $\Theta_\downarrow, \Theta_\uparrow, \Theta_\leftarrow$ and $\Theta_\rightarrow$, that ensure the safe operation of the agent during colonoscopy, as depicted in Fig. 6.4. These safety properties are formulated using the input values $x_0, \ldots, x_{15} \in \vec{x}$ of $\mathcal{F}$, where x_0 and x_{15} denote the values of the upper-left and bottom-right squares, respectively. Additionally, the five possible actions that the agent can take are denoted as $a_0, \ldots, a_4$ and are highlighted in Fig. 6.3a.

The precondition $\mathcal{P}$ is expressed using hyper-rectangles, represented as intervals, with one interval assigned to each possible input value. There are two types of intervals used to encode $\mathcal{P}$: the interval $[0, 0.6]$ denotes a safe image area that is free of obstacles, while the interval $[0.8, 1]$ represents a bright image area, indicating that the agent is in close proximity to the colon wall (as shown in Fig. 6.4). The postcondition $\mathcal{Q}$ requires the agent to select any action other than a_i, which corresponds to the unsafe action of scope motion towards the illumination direction. Therefore, to verify these properties, a FV tool searches for a single input $\vec{x}$ that satisfies $\mathcal{P}$ and for which $\mathcal{F}$ satisfies the negation of the postcondition, i.e., a configuration in which the agent selects the unsafe action a_i. If no such configuration is found, then the original property holds.

It is crucial to note that the safety properties we have defined in our study indicate the action that the agent should avoid in an unsafe situation. However, they do not specify the alternative action that should be taken instead. This is an essential aspect since it ensures that the agent is not restricted to a specific action, which may limit its ability to discover innovative and optimal strategies. Rather, the focus is solely on preventing the most harmful actions.

6.5 Experimental Validation

In this section, we present the results of the empirical evaluation of the proposed framework. The experiments aim to address the following research questions: (Q1) How does a constrained approach affect the training and performance of the agent in autonomous colon navigation? (Q2) Can minimizing violations of soft constraints lead to the elimination of hard constraint violations?

6.5.1 Experimental setup

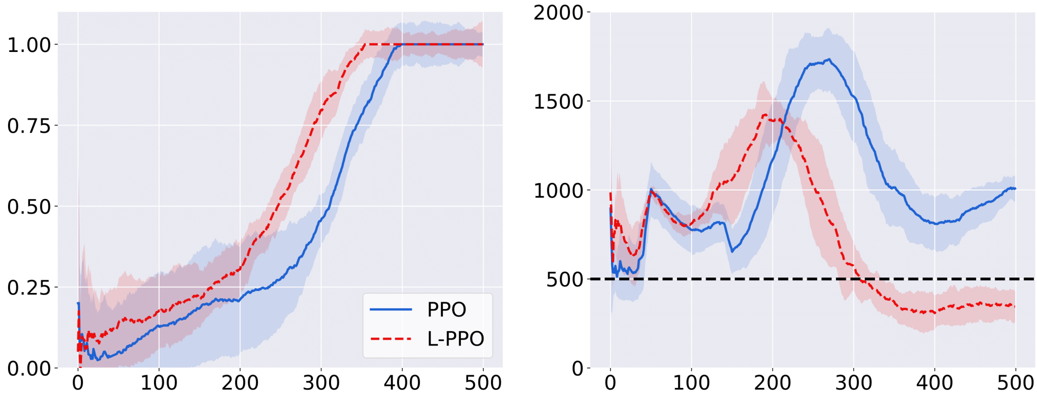
The evaluation of the proposed framework is based on four colon models of varying complexity, which were used in Chapter 5. These models have varying complexity characterized by their length and the number of acute bending angles that exceed 90°.

The primary objective of the evaluation of the proposed framework is to assess the differences in training between the proposed CRL (L-PPO) and standard DRL (PPO) approaches. The evaluation was carried out using the following steps. (S1) 5000 policies were trained on the hardest colon model using different random initialization. (S2) The best 300 policies were selected based on their success rate during training, which is the number of times the agent successfully reaches the colon end in 100 consecutive trials while minimizing the number of collisions with the walls. (S3) These 300 policies were evaluated on other colon models, and the navigation performance based on the average distance traveled by the scope on each colon model was recorded. (S4) FV was performed on the 300 policies to obtain a policy that shows no safety violation for final deployment. All data were obtained using an RTX 2070 graphics board and an i7-9700k processor.

While carrying out S3, in addition to the considered methods, we also implement PPO_{lum}. PPO_{lum} is a baseline PPO method that was trained using a lumen centralization reward function proposed in Chapter 5. The average distance traveled by the endoscope tip is a significant factor in evaluating trajectories as multiple backward motions or reversing the direction of motion may lead to suboptimal trajectories. The measurement of the distance traveled was performed using the position values of the endoscope tip that were normalized by the centerline distance of the colon model.

Table 6.1 Average distance traveled results.

	Colon 0	**Colon 1**	**Colon 2**	**Colon 3**
PPO$_{lum}$ [234]	**0.84**	0.85	0.97	0.92
PPO	0.86	0.92	0.99	0.91
L-PPO	0.88	**0.81**	**0.92**	**0.84**



(a) Expected returns

(b) Cumulative Cost

Fig. 6.5 Average performance vs the number of episodes of PPO and L-PPO over ten seeds (a) Average reward and (b) cumulative cost. Solid blue and red dashed lines are the empirical mean, while shaded regions represent the standard deviation. Black dashed line is the cost threshold

6.5.2 Training results

Fig. 6.5a presents the learning curves of PPO and L-PPO, indicating a similar performance between the two algorithms, with both achieving high reward values at approximately 400 episodes. The results demonstrate that L-PPO successfully enforces constraints by maintaining a constraint cost below the limit value at 300 episodes, while PPO's constraint cost remains above the limit. It is important to note that a single collision can produce a large number of interactions, depending on the number of timesteps the agent remains in contact with the wall. These findings suggest that, on average, L-PPO may achieve better constraint satisfaction than PPO.

The results of average distance traveled is shown in Table 6.1. Our examination shows that all three algorithms, including PPO$_{lum}$, achieve a 100% success rate in navigating all colon models by reaching the end of the colon. These results suggest that DRL can be trained without a lumen centralization reward, by using a global objective of reaching the colon end.

Table 6.2 Results of model selection. SAT indicates property violation

	Safety Properties				
	$\Theta_{\uparrow}$	$\Theta_{\downarrow}$	$\Theta_{\leftarrow}$	$\Theta_{\rightarrow}$	**Model Selection**
Method	SAT	SAT	SAT	SAT	*Completely safe model*
PPO	300	246	80	167	0
L-PPO	221	198	53	161	**3**

Additionally, the distance traveled by the scope normalized by the centerline distance of the colon model was recorded for all policies. The obtained values suggest that a shorter path than the centerline is followed.

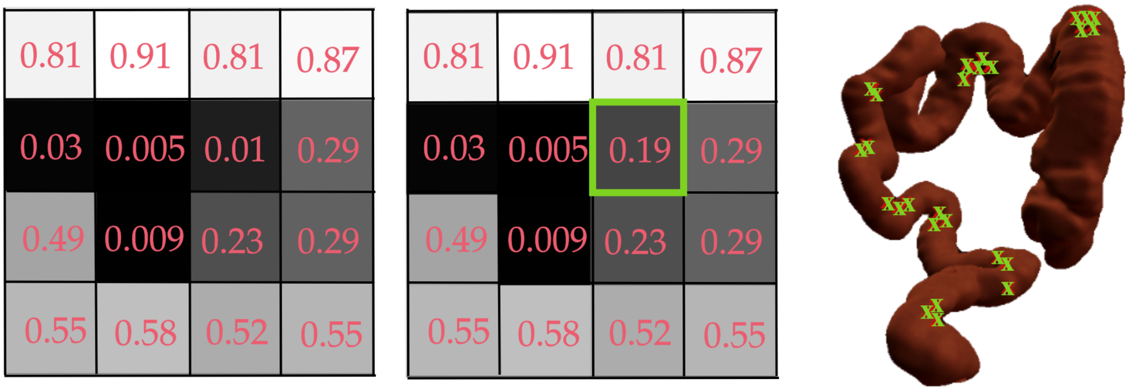
6.5.3 Formal verification results

To address Q2, FV is conducted on the 300 policies trained using each methodology. Table 6.2 provides the violations for PPO and L-PPO across all four safety properties. Specifically, for each safety property, the table reports the SAT values indicating the number of models that violate that particular property. Notably, it is observed that for the first safety property $\Theta_{\uparrow}$, which pertains to the situation where the upper part of the image is very bright and does not require an upward action from the agent, all 300 PPO policies violate the safety property. The observed violation is not straightforward to interpret, and it may be attributed to the infrequent exposure of the agent to such setups during the training process. It is plausible to suggest that the lack of sufficient training data for these specific scenarios may have hindered the agent's ability to learn the corresponding actions that adhere to the prescribed safety property.

The results presented in Table 6.2 demonstrate that L-PPO has fewer violations compared to PPO, providing evidence that the inclusion of soft constraints during training leads to a reduction in hard constraint violations. Fig. 6.6c illustrates the locations of the hard constraint violations for PPO on one of the colon models. The violations occur predominantly at sharp bends, which are crucial points for the correct execution of the colonoscopy procedure.

This analysis aims to determine if it is feasible to identify a policy that adheres to all the hard constraints. As evidenced by Table 6.2, three models satisfying all the hard constraints were identified in the case of L-PPO, while no policies conforming to the same standards were observed in the case of PPO, thus demonstrating the effectiveness of the proposed framework. It is worth noting that the L-PPO utilized in prior experiments (such as Table 6.1) is one of the three safe policies.

The vulnerability of DNNs and the necessity of using FV in safety-critical scenarios are emphasized by Fig. 6.6. Specifically, Fig. 6.6 displays the results of the analysis on a policy trained with PPO, wherein Fig. 6.6b shows an input on which the tested model acts safely, without violating the property, while an adversarial input is discovered by the formal verifier



(a) (b) (c)

Fig. 6.6 (a)Adversarial example discovered with FV for the safety property $\Theta_{\uparrow}$. (b) A small perturbation in the square marked green, the agent shows safe behavior. (c) Hard constraint violation positions for one of the PPO policies marked with green crosses.

with the same property $\Theta_{\uparrow}$ in Fig. 6.6a. The figure illustrates that the input only differs by 0.1 in the 6th value; however, this insignificant alteration causes the network to output a secure action in one case and a potentially dangerous action in the other, underscoring the criticality of utilizing FV to ensure safety in DNN-based systems.

6.6 Conclusions

We have investigated the challenges associated with the deployment of DRL for autonomous colonoscopy navigation in a virtual simulation. DRL-based methods have demonstrated the ability to successfully traverse patient-specific colon models with comparable performance to that of expert clinicians [234]. Nevertheless, these methods are susceptible to adversarial attacks, which could result in safety violations with potentially fatal consequences. Consequently, we exploit a CRL approach that ensures soft safety constraints through a cost function of safety violations. However, enforcing hard constraints through this methodology is difficult. To this end, we propose a model selection strategy that harnesses FV to evaluate the safety of a vast pool of policies trained using CRL. The FV is a modular framework capable of verifying any given set of safety properties and is able to provide guarantees of safe behavior prior to deployment. From the 300 policies trained using CRL, we identified three policies that adhered to all safety constraints, compared to no policies that met the same criterion for standard DRL.

Contributions of this chapter

1. An end-to-end DRL method for colon navigation that eliminates the need for a separate lumen detection system.
2. A CRL approach that constrains the policy in a pre-defined safe state-space to minimize potentially dangerous actions.
3. A model selection strategy that selects policies satisfying all safety constraints, with each policy formally verified to check for its safety violations.

Publications linked to this chapter

1. Davide Corsi*, Luca Marzari*, Ameya Pore*, Alessandro Farinelli, Alicia Casals, Paolo Fiorini, and Diego Dall'Alba. "Constrained Reinforcement Learning and Formal Verification for Safe Colonoscopy Navigation." arXiv preprint arXiv:2303.03207 (2023). * - Equal contribution

Part II

CRC detection

Chapter 7

CRC diagnosis using OCT scanning

7.1 Introduction

Colorectal polyps are widely recognized as a significant precursor to CRC. Polyps can be classified into neoplastic and non-neoplastic polyps based on their features such as color, shape, texture, size, borders, and vessels. The size of polyps plays a crucial role in determining the degree of malignancy, with larger polyps posing a higher risk [56]. Early detection of diminutive and small polyps is essential to reduce the risk of metastasis.

The current standard for CRC diagnosis is colonoscopy screening, where a flexible endoscope is used for visual inspection of the colon walls [83]. In case of suspicion of precancerous tissue, a biopsy is performed, and the tissue sample is sent for histological analysis. If the results indicate a high probability of CRC percussion, the polyp is removed using specific polyp removal procedures such as polypectomy, endoscopic mucosa resection, or endoscopic submucosa dissection.

However, visual inspection alone is not sufficient for early detection of small or diminutive polyps, leading to missed early-stage malignancies [311, 312]. Additionally, visual inspection also reduces the efficacy of the screening after polyp removal, as clumps of tumor cells may persist beneath the mucosal layer [313]. To address these limitations, an enhanced imaging modality is needed to improve the efficacy of CRC surveillance.

OCT offers a high-resolution cross-sectional imaging approach for the characterization of polyps [57]. This imaging modality is non-invasive and provides near-microscopic resolution images of tissue with millimetric penetration depth, which minimizes the necessity of tissue removal and ex-situ biopsies. Several studies have demonstrated that OCT accurately differentiates normal from abnormal colonic tissue with promising success [57].

Robotic FE platforms such as STRAS [314] provide the opportunity to mount external sensors like OCT probes on the distal part of the endoscope bodies [315] (as depicted in Fig. 7.1). OCT probes for endoscopy require miniaturization of optics, leading to limited field of view and depth perception of around 5 mm. Thus, to expand the field of view, the

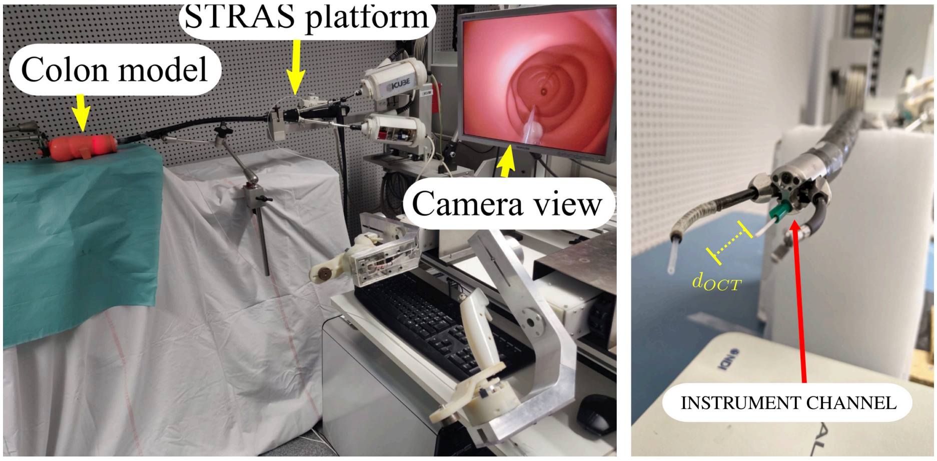


Fig. 7.1 (a) STRAS robotic setup. (b) Motorized OCT probe.

operator must manually scan the area of interest using the probe while maintaining contact with the tissue, which increases the risk of missing lesions and do not ensure correct scanning pattern [315]. In order to perform the scanning task, the operator must control multiple DoFs, while relying on both the endoscopic camera and the OCT images, which is difficult even in telemanipulation mode. This telemanipulation task is challenging since the conventional monocular camera on the endoscopic robot can only provide 2D images of the tissue surface and the OCT probe. Moreover, it is difficult to estimate the distance between the tissue and the probe using OCT images when the distance is beyond the perception range. Therefore, autonomous scanning via IBVS will benefit clinicians by extending the OCT field of view and reducing their physical workload, allowing them to concentrate on real-time diagnosis. Due to the relatively large surface area of the colon wall compared to the working space of the OCT probe, an efficient scanning strategy is necessary to cover the region of interest within the time constraints of the procedure.

This study proposes a novel autonomous scanning approach for real-time polyp diagnosis in the colon. The scanning process is segmented into four subtasks as follows:

S_1 : Eye-in-hand IBVS - The control mechanism used in this task is IBVS, which moves the endoscope body towards the lumen center using visual information. The detailed explanation of the endoscope control mechanism is given in Sec. 7.4.

S_2 : Eye-to-hand IBVS - This subtask entails controlling the OCT probe to reach the visually detected polyp. The control of the OCT probe is achieved through IBVS. The specifications of the OCT probe and the approach towards the polyp are provided in Sec. 7.5.2 and Sec. 7.5.3, respectively.

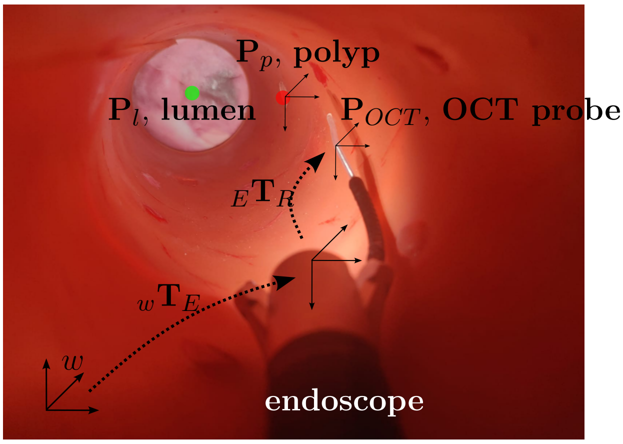


Fig. 7.2 Overview of experimental setup depicting the realistic colon model, the lumen, OCT probe and the endoscope body.

S_3 : OCT Scanning - In this subtask, the OCT probe is maneuvered to scan the area of the polyp. The movement of the OCT probe is determined based on the information obtained from the previous subtasks.

S_4 : Polyp Assessment - This subtask involves image-based inference to classify the health of the polyp in real-time. The classification of the tissue is based on the information obtained from the OCT scans. The detailed process of tissue classification can be found in Sec. 7.5.5.

In each subtask, multiple objectives might be present, and these objectives are defined under different reference frames. To address this issue, we adopt the Hierarchical Quadratic Programming (HQP) formulation of the optimization function for each objective, described in Sec. 7.3.2. The efficacy of this approach is evaluated through tests conducted on a realistic colon model [316], which simulates visual and surface tissue features. Our results demonstrate that this approach can perform real-time scanning of the tissue surface to assess the health of tissue from normal to precancerous stages.

7.2 Related Works

Eye-in-hand IBVS: In recent years, several works have explored the development of robot-assisted colonoscopy navigation, which aims to provide a more efficient and accurate method for colon examination. The primary objective of these works is to assign the target direction of the endoscope motion towards the estimated center of the lumen. To achieve this objective, various techniques have been used, including lumen centralization control strategies described earlier in Chapter 5 and Chapter 6 [178, 54, 294, 234, 296].

Eye-to-hand IBVS: Martin *et. al.* [317] and Zhongkai *et. al.* [179] developed an autonomous biopsy method for endoscopic procedures where the tool channel projection is aligned with the tissue target. Both of these approaches employed the use of multiple viewpoints to estimate the position and depth of the target tissue. In an effort to reduce the distance error between the target tissue and the probe, Zhang *et. al.* [318] proposed a marker-based OCT probe detection approach. The OCT probe stops at the contact detection point, thus reducing the risk of tissue damage during the procedure. However, the use of artificial fiducial markers can be a limitation, as it requires manual placement of the markers and may interfere with the visual inspection of the tissue. To address this limitation, our proposed approach is marker-less OCT probe detection using image segmentation with CNNs. By eliminating the need for markers, the system is able to achieve greater accuracy and flexibility, allowing for a more effective implementation of the Eye-to-hand IBVS method in endoscopic procedures.

OCT Scanning: Obtaining volumetric information using OCT involves rotating and pulling back a side-focused optical probe inside a static sheath. The cylindrical area defined by the length of the pullback stroke and the working distance of the optics determines the optical scanning area [319]. Although manual scanning can be performed by moving the endoscope, it has limited capabilities in terms of sampling uniformity, leading to an irregular pattern [320]. The commonly used OCT scanning patterns, such as circular, spiral, and raster scanning, cannot be employed in an endoscopic OCT probe due to the miniaturized optics that reduce the field of view. As OCT collects only one-dimensional information in a single measurement, scanning is necessary for obtaining volumetric information. Several previous works have proposed to move the tissue for scanning [321–323], but this approach is not feasible in colonoscopic scenario. Recently, a study by Zhang *et al.* [318] proposed an OCT scanning approach based on a designed 3D curvature trajectory. However, this approach has the disadvantage of low overlapping of information between sequential scans, making volumetric reconstruction more difficult. To address these limitations, we propose an automatic scanning strategy using a steerable OCT catheter to improve imaging performance while scanning areas larger than the field of view of a low-profile OCT probe.

Polyp assessment : OCT has been demonstrated to effectively distinguish between normal and abnormal tissue in various organs, providing an optical biopsy-like approach in both human and murine colorectal models [324, 325]. However, the implementation of this technology in a clinical setting is challenging due to the large volume of data generated and the intricate qualitative variations between normal and abnormal tissue [57]. The study by Zeng *et al.* [57] was the first to demonstrate real-time in-situ characterization of polyps through manual OCT scanning. Similarly, in this study, we employed a method for polyp assessment utilizing autonomous OCT scanning.

7.3 Optimization based control formulation

Optimization-based control formulation is a mathematical framework used to design controllers for complex systems with multiple objectives and constraints. It involves finding a control law that minimizes a cost function subject to constraints on the system's dynamics, input/output variables, and physical limitations. We use the STRAS robotic system to illustrate the control formulation.

7.3.1 System Specification

The STRAS robotic platform [326] was designed for performing intraluminal endoscopic procedures, such as ESD. It consists of a flexible endoscope that allows the user to perform telemanipulation with a camera and instruments (arms) at its tip (Fig. 7.1. The endoscope has four DoFs and is capable of horizontal and vertical bending, rotation, and translation along its axis. The endoscope body has three channels, with two of them equipped with surgical instruments and one for fluid management.

The Constant Curvature Model (CCM) is used to represent the flexible segments of the robotic structure [326]. The configuration variable $\mathbf{q}$ is set as $\mathbf{q} = [\mathbf{q}_R, \mathbf{q}_E]^T$, with $\mathbf{q}_R$ representing the configuration variables of the OCT probe set in the right channel of the endoscope body and $\mathbf{q}_E$ representing the configuration variables of the endoscope. The configuration variables for the endoscope can be expressed as $\mathbf{q}_E = [\beta_{E_h}, \beta_{E_z}, \alpha_E, t_E]$ which describe the horizontal and vertical bending, rotation, and translation, respectively.

The lateral and vertical displacement of the endoscope is limited to ±5 cm. For such small displacements, the camera motion can be modeled as movement in the x-y plane. The Homogeneous Transformation Matrix (HTM) $w\boldsymbol{T}^E$ describes the endoscope tip position with its translational component, ${}_w\mathbf{p}_E = pos(w\boldsymbol{T}^E)$. Here, w refers to the world frame of reference while E refers to the endoscope frame of reference. The endoscope Jacobian ${}_w\mathbf{J}_E \in \mathbb{R}^{3x4}$ is taken from prior camera displacement measurements in a controlled scenario [318] such that the endoscope velocity (${}_w\dot{\mathbf{p}}_E$) is defined as:

$${}_w\dot{\mathbf{p}}_E = w\mathbf{J}_E\dot{\mathbf{q}}_E \tag{7.1}$$

The endoscope houses the arms, such that motion of the endoscope impacts arms position in the world frame. Yet, arm motion are independent of the body in the endoscope frame. Similarly to the endoscope kinematics model, the OCT probe is modeled using the CCM, as in prior implementations of the STRAS platform for OCT probe control. The tip position of the probe with respect to the endoscope camera frame is described by the translational component of the HTM, ${}_E\mathbf{p}_R = pos({}_E\boldsymbol{T}^R)$. The OCT probe is described as $\mathbf{q}_R = [\beta_R, \alpha_R, t_R]$ with β being the OCT probe bending, α the rotation and t the translational component, such that the Jacobian $J_R \in \mathbb{R}^{3x3}$ is set as:

$$ {}_{E}\dot{\mathbf{p}}_{R} = {}_{E}\mathbf{J}_{R}\dot{\mathbf{q}}_{R} \tag{7.2} $$

7.3.2 Hierarchical Quadratic formulation

The optimal control velocity of the system, $\dot{\mathbf{p}}^{\star}_{m}$ is denoted as the combination of the optimal velocity of the endoscope tip, $\dot{\mathbf{p}}^{\star}_{E}$ and the OCT probe, $\dot{\mathbf{p}}^{\star}_{R}$, described by the following equation:

$$ \dot{\mathbf{p}}^{\star}_{m} = \begin{bmatrix} {}_{w}\dot{\mathbf{p}}^{\star}_{E,m} \\ \\ {}_{E}\dot{\mathbf{p}}^{\star}_{R,m} \end{bmatrix} \quad , \quad m \in [S_1, S_2, S_3, S_4] \tag{7.3} $$

where m denotes the subtasks in the range of $[S_1, S_2, S_3, S_4]$, and the velocity of each subtask is determined by the target position. To achieve independent speeds for each XYZ component in the two controllable subsystems (OCT probe arm and endoscope), the current axis position $(\cdot)t$ and the desired axis position $(\cdot)d$ are used to set each component velocity, as described below:

$$ \dot{p}^{\star}_{m,u} = (p_{m,u,t} - p_{m,u,d})k_{m,u} \quad , \quad u \in [x, y, z] \tag{7.4} $$

where k_u is a scalar gain with units $[\frac{1}{s}]$. Since each desired velocity is represented in a different reference frame, we stack the desired velocities for each subtask in the general Jacobian as:

$$ \mathbf{J} = \begin{bmatrix} {}_{w}\mathbf{J}_{E} & 0 \\ 0 & {}_{E}\mathbf{J}_{R} \end{bmatrix} . \tag{7.5} $$

At each m subtask, different desired Cartesian velocities can be set, the required joint velocity is computed by solving the optimization formulation:

$$ \min_{\dot{\mathbf{q}}} \|\mathbf{J}_{m}\,\dot{\mathbf{q}} - \dot{\mathbf{p}}^{\star}_{m}\|_{2} \, . \tag{7.6} $$

Where the underscript term m present on the Jacobian and the desired speed denotes a specific formulation per each subtask, effectively modifying the system behavior. Expression (7.6) can be formulated as a Quadratic Programming (QP) problem [318]. Constraints and limits are introduced in the QP form allowing to effectively implement collision avoidance and reducing the movement towards joint limits, such that:

$$ \min_{\dot{\mathbf{q}}} \frac{1}{2}\dot{\mathbf{q}}^{T}\mathbf{H}\dot{\mathbf{q}} + \mathbf{c}^{T}\dot{\mathbf{q}} \tag{7.7} $$

$$ s.t. \quad \mathbf{A}\dot{\mathbf{q}} \leq \mathbf{b} \tag{7.8} $$

$$ \dot{\mathbf{q}}_{min} \leq \dot{\mathbf{q}} \leq \dot{\mathbf{q}}_{max} \tag{7.9} $$

where the $\mathbf{H} = \sum_{m} \gamma_m H_m$, $\mathbf{c} = \sum_{m} \gamma_m c_m$, and γ_m is a term that specify the hierarchy of the m subtask. Each semi-positive matrix $\mathbf{H}_m$ is formalized as $\mathbf{H}_m = \mathbf{J}_m^T \mathbf{J}_m$ with $\mathbf{J}_{\{6x7\}}$, and the $\mathbf{c}_m$ vector defined as $\mathbf{c}_m = -k_m \mathbf{J}_m^T \dot{\mathbf{p}}_m^{\star}$, with k_m a proportional gain for the subtask of units $[\frac{1}{s}]$. The optimization variable is set as $\dot{\mathbf{q}}_{\{7x1\}} = [\dot{\mathbf{q}}_R, \dot{\mathbf{q}}_E]^T$. Different γ_m values are defined for each m task; if a subtask is not active, $\gamma_m = 0$ nullifying such substask. When $\gamma_m \neq 0$ the solver [327] will compute a $\dot{\mathbf{q}}$ that gradually reaches the multiple velocity objectives. An overview of how each Jacobian $\mathbf{J}_m$ and desired velocities $\dot{\mathbf{p}}_m^{\star}$ is formulated per each m subtask is presented on the following sections.

7.4 Endoscope Control

The aim of the endoscopic visual servoing is to center the lumen in the image. The image-detection module provides the image position of the polyp, represented by $\mathbf{p}_p$.

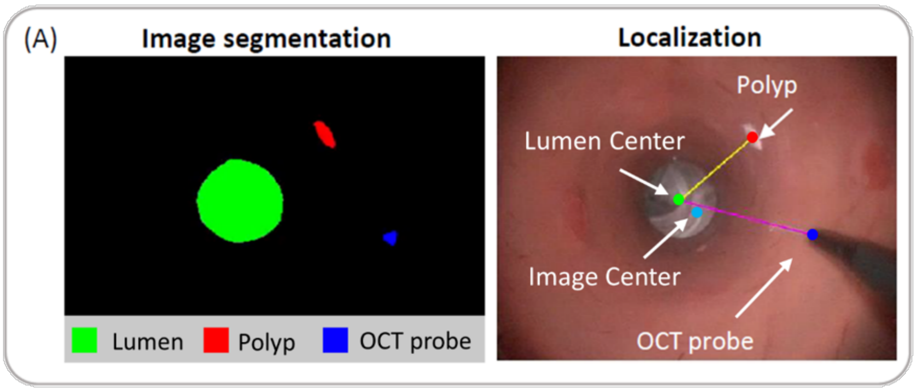


Fig. 7.3 Global navigation with information from endoscopic camera. (A) Real-time endoscopic image segmentation for lumen, polyp and OCT probe localization.

7.4.1 Image position control

Lumen alignment is performed to complete subtask S_1 by matching lumen image position $\mathbf{p}_l = (p_l^x, p_l^y)$ to normalized image center $\mathbf{p}_c = (p_c^x, p_c^y)$. Note that the OCT probe is not deployed in this subtask. Therefore the control velocity required for lumen alignment is determined as follows:

$$\dot{\mathbf{p}}_{S_1}^{\star} = \begin{bmatrix} {}_{W}\dot{\mathbf{p}}_{E,S_1}^{\star} \\ \\ {}_{E}\dot{\mathbf{p}}_{R,S_1}^{\star} \end{bmatrix} \quad , \quad {}_{W}\dot{\mathbf{p}}_{E,S_1}^{\star} = \begin{bmatrix} (p_l^x - p_c^x)\, k_{x,S_1} \\ (p_l^y - p_c^y)\, k_{y,S_1} \\ 0 \end{bmatrix} \quad , \quad {}_{E}\dot{\mathbf{p}}_{R,S_1}^{\star} = \begin{bmatrix} \mathbf{0}_{\{3x1\}} \end{bmatrix} \tag{7.10}$$

with $k_{u,S1}$ being the specified gain per each point component. The values of p_c^x and p_c^x are assigned as 0.5. While the task Jacobian is set as:

$$\mathbf{J}_{S_1} = \begin{bmatrix} 0 & 0 \\ 0 & {}_{w}\mathbf{J}_E \end{bmatrix} \tag{7.11}$$

7.5 OCT probe control

7.5.1 Polyp detection

The detection of polyps in the phantom is accomplished through the utilization of supervised deep learning techniques. The creation of the training set was achieved by acquiring images through the telemanipulation of the robot. These images, with a resolution of 720 × 576 pixels, were manually annotated by identifying the polyp region in the images, which were then differentiated from the background. To enhance the robustness of the training process, the 100 collected images were augmented. A U-NET architecture [328] was implemented and trained using a supervised approach on the annotated image dataset. The output of the trained U-NET was utilized to estimate the center of mass of the detected polyp, denoted as $\mathbf{p}_p$.

7.5.2 Probe specification and model

In this study, we utilize planar radial B-scan images that are obtained through the integration of an endoscopic OCT catheter with a diameter of 3.5 mm [315] into the instrument channel of the STRAS [326]. The distal end of the instrument is equipped with a transparent sheath, allowing for three-dimensional OCT imaging using a side-focusing optical probe that is rotatable and has two proximal external scanning actuators. The OCT imaging system is built around the Axsun engine and includes a 1310 nm center wavelength swept source laser and 100 kHz A-line rate.

The OCT catheter is compatible with the instrument channel of a robotized flexible interventional endoscope and the resulting OCT image stream is stabilized using a CNN based method [329, 330]. After stabilization, the image is segmented to calculate the distance and direction between the scanning center and the surrounding tissue.

7.5.3 Polyp approach by OCT probe IBVS

The OCT probe position $\mathbf{p}_{OCT} = (p_{OCT}^x, p_{OCT}^y)$ is set to match image polyp position $\mathbf{p}_p = (p_p^x, p_p^y)$ while being within the OCT detection range to perform pull-back. The objective velocities are defined as:

$$\dot{\mathbf{p}}^{\star}_{S_2} = \begin{bmatrix} {}_{w}\dot{\mathbf{p}}^{\star}_{E,S_2} \\ \\ {}_{E}\dot{\mathbf{p}}^{\star}_{R,S_2} \end{bmatrix} \quad , \quad {}_{w}\dot{\mathbf{p}}^{\star}_{E,S_2} = \begin{bmatrix} \mathbf{0}_{\{2x1\}} \\ \\ \mu_1 \end{bmatrix} \quad , \quad {}_{w}\dot{\mathbf{p}}^{\star}_{R,S_2} = \begin{bmatrix} (p_p^x - p_{OCT})\, k_{x,S_2} \\ (p_p^y - p_{OCT})\, k_{y,S_2} \\ 0 \end{bmatrix} \tag{7.12}$$

where μ_1 is a constant term to effectively translate the endoscope while aligning with the lumen. When the OCT probe is close to the tissue position, the endoscope translation is stopped i.e. if $z(p_{OCT} - p_p) \leq \eta_z$, $\mu_1 = 0$. $\eta_z = 4$mm is the threshold we setup for the closeness with the OCT detection range set to 5mm. When the distance between the OCT probe and polyp is above the threshold, $\mu_1 \approx 2$mm/sec. Currently, we consider polyp position in quadrant 1 of the endoscopic image for this preliminary study where the entire quadrant is reachable by the probe.

7.5.4 OCT contact control

After S_2 is completed, subtask S_3 is initiated. We have proposed a scanning strategy that uses multiple parallel translational pullbacks (as depicted in Fig. 7.4**c**). This method offers several advantages. Firstly, the relative longitudinal position between the reference object (i.e. the protective sheath) and the rotation lens is fixed, providing stability for the OCT imaging. Secondly, this scanning strategy allows for the acquisition of a stack of B-scan slices that are highly aligned with each other for each pullback, reducing the need for correction. If further volumetric reconstruction is required for diagnostic purposes, a volumetric stitching algorithm [331] can be used to connect the small volumes obtained from each pullback.

The multi-pullback strategy was chosen over the swiping pullback strategy due to the shape of the colon lumen. After gas inflation, the colon lumen maintains a certain level of cylindrical shape. However, swiping the instrument along the inner surface of the lumen can result in large instrument displacement. On the other hand, translational pullback is well-suited for cylindrical lumens regardless of size, and moving along the lumen wall requires less compensation of displacement during the scanning process. Algorithm 1 provides the meta-code for the OCT scanning strategy.

7.5.5 OCT tissue health classification

Drawing inspiration from the work presented in [57], which demonstrated the ability to differentiate healthy colon tissue from pathological tissue using layer features obtained from cross-sectional images acquired using OCT, we have sought to realize the identification of unhealthy colon tissue based on a previously proposed layer contour segmentation algorithm [332]. As depicted in Fig. 7.4-e, the algorithm processes OCT images in the polar domain to determine the positions of tissue layers and outputs the classification results based on the layer segmentation.

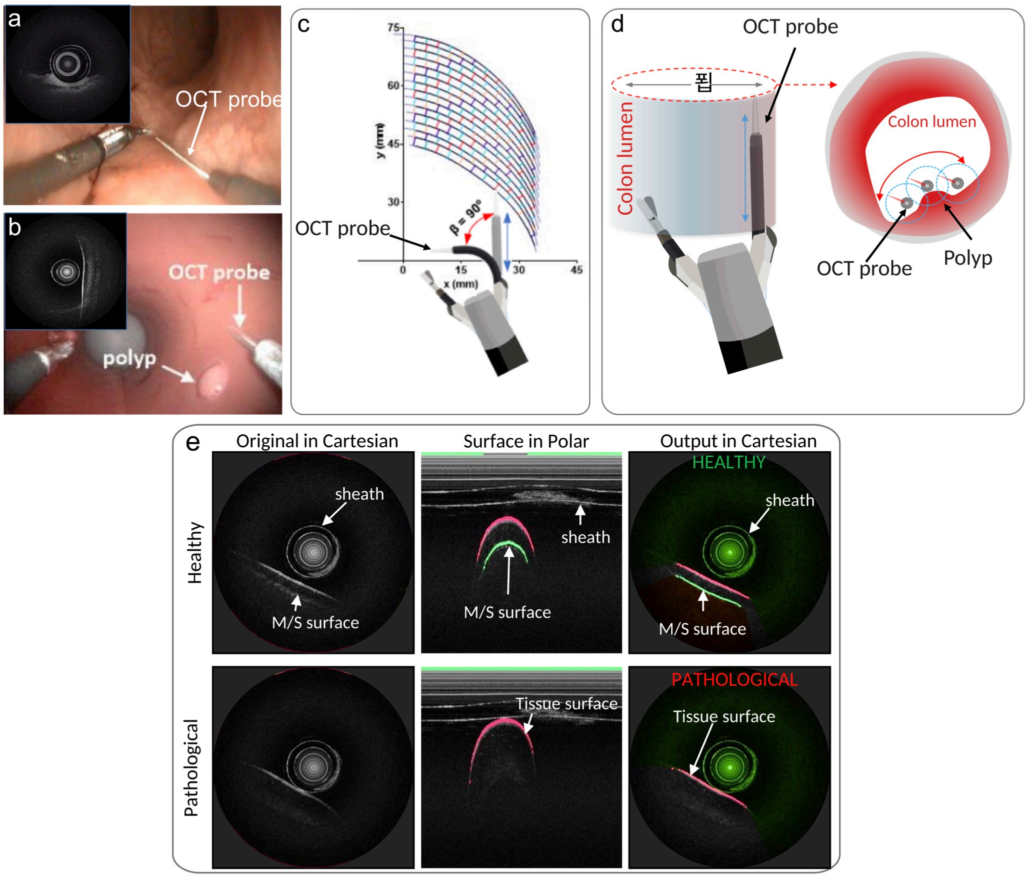


Fig. 7.4 Scanning strategies for colon lumen with robotized endoscopic OCT. (a) shows our system manually operated within a colon, which is one of our previous work on OCT/STRAS integration [315], and in (b) we rotate the colon tissue phantom to simulate such lumen environment. (c) shows OCT swiping pullback following a repeat arc trajectory, and (d) shows another scanning strategy that utilizes multiple parallel translation pullbacks. (e) Classify healthy/unhealthy tissue using OCT images. OCT images are processed in polar domain for multi-surface segmentation and reconstructed in Cartesian for display. In healthy tissue muscle/submucosa layers (M/S) have clear boundary[316], while in unhealthy tissue submucosa layer disappears.

7.6 Experiments

To verify the validity of the proposed HQP control approach, five trials were conducted for polyp evaluation.

7.6.1 Overview of the implementation

The workflow of the experiments is presented in Fig. 7.5. The STRAS platform is randomly positioned within the synthetic colon model, due to the limitations in translation on the STRAS system which was designed to prioritize complex intraluminal surgical gestures rather

Algorithm 1 OCT local scanning

```
while true do
    Obtain translation position state $t_k$
    assign probe translation $\Delta t$:
    if reach distal limit then
        $t_{k+1} = t_k - \Delta t$
    else
        $t_{k+1} = t_k + \Delta t$
    Set target contact as $c_t$
    Compute distance value $d_k$ and contact region size $c_k$ from OCT image
    fix bending $b_x = 0$, obtain outward bending state $b_{y,k}$
    Compute contact error $E_k = c_t - (c_k - d_k)$
    Compute new target bending $b_{y,k+1} = b_{y,k} + P(E_k) + D(E_k)$
    set target translation $t_{k+1}$, and bending $b_x$, $b_{y,k+1}$ to STRAS follower
```

than intraluminal navigation. The limits of the platform were modified for each subtask in the experimental setup.

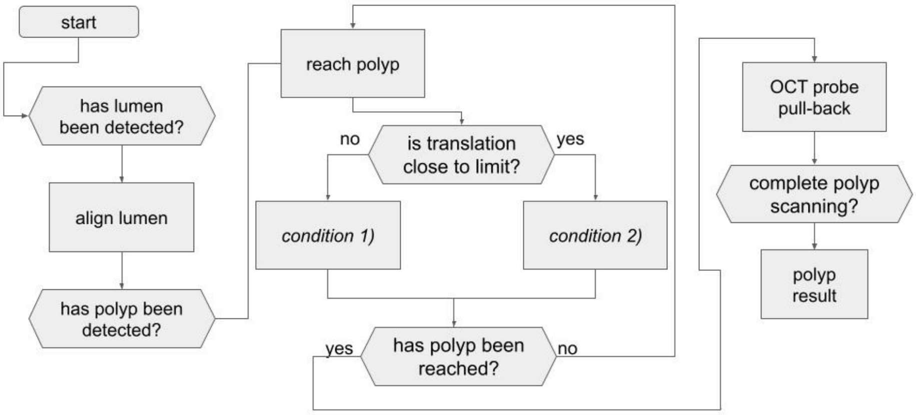
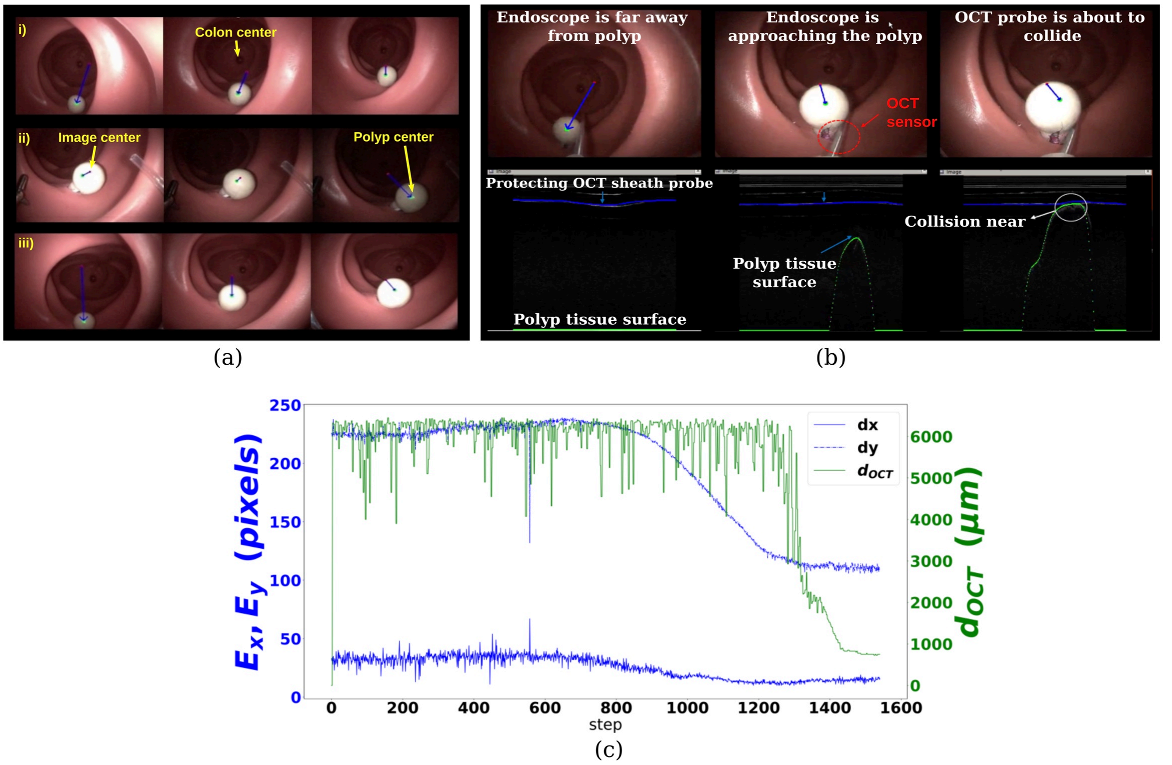


Fig. 7.5 Flowchart of the autonomous scanning and tissue classification workflow

7.6.2 Results

We conducted our experiments on the ascending colon section of the LM-107 Colonoscopy Simulator (KOKEN, Japan), which contains a white polyp that can be placed in a predetermined location. Our main goals were twofold: first, to demonstrate the robustness of polyp detection in different lighting conditions, and second, to employ an OCT probe for tissue scanning.

To achieve the first goal, we implemented the image position control method outlined in the previous section and evaluated polyp detection under varying luminosity conditions, as



(a) (b)

(c)

Fig. 7.6 a) Image guided control. Sequence of frames obtained from the monocular camera while aligning towards the polyp (i) in normal lighting conditions, (ii) in varying lighting conditions, (iii) Experiment two: translating and aligning towards the detected polyp. b) Top: Deployment of the OCT probe. Bottom: Data generated while tissue scanning. c) Plot of evolution of E_x, E_y and d_{OCT} for the second experiment performed.

shown in Fig. 7.6a. It is important to note that the OCT probe was not utilized during these experiments.

Our second objective is to perform simultaneous tasks of aligning with the center of the lumen, S_1, and deploying the OCT probe to reach the polyp, S_2. To achieve this, we implement the image position control and translation experiments as illustrated in Fig. 7.6a. The distance between the center of the image (red dot) and the center of mass of the detected polyp (green dot) is shown in blue.

In Fig. 7.6b (bottom), the polyp surface detection in the OCT image and the corresponding endoscopic image in the top row are shown. The green line represents the polyp surface, while the blue line shows the sheath of the OCT catheter probe. When the endoscope is not aligned with the polyp, both lines remain straight, as the polyp is outside the field of view of the OCT probe. However, as the endoscope approaches the polyp, a peak point occurs in the green line, indicating the presence of an object near the probe. The peak point grows larger as the endoscope advances towards the polyp, and finally touches the blue line when the probe is about to collide with the polyp. Thus, the feedback from the OCT image prevents collision.

The plot for the error in the experiments is shown in Fig. 7.6c. We set a safety buffer of 10 seconds (830 steps) for visual inspection at a step rate of 50Hz. After this buffer period, the pixel error E starts reducing, and the d_{OCT} also starts decreasing after 1300 steps, indicating that the probe has come close enough to the polyp to provide a measurement with less error.

Subsequently, we executed an automatic scanning process utilizing a predefined multi-pullback approach and assessed the polyps. To achieve this, fake polyps were created by utilizing *dragonskin* ecoflex that disrupted the underlying tissue layers. We observe that our system outputs 100% accuracy in identifying such abnormalities.

7.7 Conclusion and Future work

In conclusion, our research demonstrates the potential of steerable endoscopic OCT catheter robotization for autonomous scanning of malignant tissue. We have presented a comprehensive strategy that integrates lumen and polyp detection, navigation following the lumen centerline, probe alignment to reach the polyp, tissue scanning and assessment in real-time.

The potential benefits of implementing automatic scanning in clinical settings are significant, as it would allow the clinician to focus on medical diagnosis rather than controlling the catheter device. With free hands, the operator can stop scanning at a particular point to examine the area more closely without losing track of the current position.

There is however, one major limitation that the chosen robot model is based on a constant curvature assumption. This model does have limitations, including a lack of compliance with the actual robot's behaviour, which may result in deviations from expected performance.

Future work will focus on increasing the complexity of the testbed to demonstrate more robust control, using scanning volume metrics to maximize the area scanned around the polyp [319], and developing a DRL strategy to optimize all control objectives in an end-to-end manner. These advancements will significantly reduce the amount of hand-engineering required for each subtask, allowing for more efficient and effective autonomous scanning of malignant tissue in clinical settings.

Contributions of this chapter

1. An autonomous scanning strategy for real-time CRC polyp diagnosis. The system integrates the following features: Eye-in-hand IBVS, Eye-to-hand IBVS, OCT Scanning and Polyp Assessment. These objectives are encoded in a QP formulation of the optimization functions.

Publications linked to this chapter

1. Herrera, Jose Fernando Gonzalez, Ameya Pore, Luca Sestini, Sujit Kumar Sahu, Guiqiu Liao, Philippe Zanne, Diego Dall'Alba, Albert Hernansanz, Benoit Rosa, Florent Nageotte and Michalina J Gora. "Autonomous image guided control of endoscopic orientation for OCT scanning." In CRAS, Naples, Italy, avril 2022. 2022.

2. Herrera, Jose Fernando Gonzalez, Ameya Pore, Luca Sestini, Sujit Kumar Sahu, Guiqiu Liao, Philippe Zanne, Florent Nageotte, Michalina J Gora, Benoit Rosa. "Robotic Autonomy for real-time colorectal cancer diagnosis using Endoscopic OCT Scanning." to be submitted, Robotics and Automation Letter.

Part III

Tissue manipulation

Chapter 8

Learning soft tissue manipulation

8.1 Introduction

Colonoscopy enables the resection of adenomatous polyps, which are known CRC precursors, through procedures like Endoscopic Mucossal Resection EMR or Endoscopic Submucosal Dissection ESD. These procedures employ dual-channel flexible endoscopes alongside passive instruments inserted into the channels [333]. However, these tools are not ideal for complex surgical procedures, as they lack the ability to independently control surgical tools and the endoscopic camera, do not enable triangulation, and are deficient in DoFs. As a result, endoscopic removal of polyps becomes challenging when they are large, flat, situated in high-risk locations, or difficult to access [334]. Inadequate retraction and an unstable view lead to a high incidence of complications, such as muscular layer perforation and high recurrence rates [335].

Therefore, the use of ESD and EMR procedures is restricted mainly to eastern countries where the prevalence of digestive cancers is high, and where endoscopists receive intensive training in flexible endoscope manipulation [336]. In contrast, in western countries, only a few experienced endoscopists perform these procedures routinely [276]. Few flexible robotic platforms have been developed which provide more DoFs and simultaneous bimanual control of instruments from comfortable master consoles such as the acstras [276] and Endosamurai [337] robotic systems. However, these platforms are still in research phase and have not received clinical certification.

Conventional robotic systems, such as the dVSS, have been used for transanal MIS [338]. The dVSS consists of three robotic arms, called PSM, equipped with articulated MIS instruments and controlled by the surgeon via a console with two master handlers.

A substantial portion of the polyp excision procedure involves mobilizing and manipulating tissue, including the grasping and lifting of a thin layer of tissue to access an underlying area [339]. This gesture, known as TR, is necessary in multiple stages of the surgery and involves interacting with soft tissues having varying physical and geometric properties, such as stiffness and viscoelasticity, which show high variability both inter and intra-subject.

When using the dVSS, TR is temporarily carried out using the third PSM or additional instruments are used controlled by an assistant operator [58]. This requires the surgeon to either switch between robotic arms during the surgery with a different set of visuomotor feedback and limited perception or instruct an assistant with the desired motion [58]. Such a protocol increases the surgeon's cognitive load, as well as the risks of tissue damage and instrument collision. As a result, automating the TR subtask can benefit surgeons by enabling them to focus on critical aspects of the surgery and potentially improve the overall outcome.

The key obstacle in automating robotic tissue manipulation is the need to account for the dynamic behavior of soft tissues interacting with the anatomical environment. Some researchers have attempted to automate TR using standard motion control algorithms. For example, Nagy *et al.* proposed an approach for TR that uses soft computing methods based on images, where three methods based on proportional control, HMMs and fuzzy logic are validated [340]. Attanasio *et al.* developed a trajectory planner based on coordinates extracted directly from intra-operative image feed [58]. However, both of these methods automated TR using pre-defined movement sequences without considering tissue dynamics, which may limit their applicability in realistic anatomical environments. In other works, such as [341, 342], researchers employed standard path planning methods, such as probabilistic roadmaps, to generate an optimal plan for the task using biomechanical simulations in the preoperative phase, accounting for the deformation properties of the anatomy. Nevertheless, all these approaches rely on hand-crafted control policies, which can make executing complex nonlinear trajectories and behaviors challenging.

Most of the prior works concerning the automation of actions involving soft tissues manipulation has relied heavily on the use of LfD, where a task is learned by imitating an expert's behavior [343, 344, 49, 345]. Reiley *et al.* proposed a LfD-based framework that uses Gaussian Mixture Models (GMM) to generate motion [346]. Recently, a similar approach using GMM has been used to learn dynamic motion primitives from demonstrations given by expert surgeons [345]. Osa *et al.* introduced an iterative technique to learn a single reference trajectory for knot tying [48]. However, a single demonstration is not sufficient to model a manipulation skill effectively. Schulman *et al.* used a trajectory transfer algorithm to learn from demonstrations for the task of suturing [343]. Murali *et al.* developed a method to segment demonstrations into motion sequences [49].

Although LfD is a preferred approach, as it allows for proper interaction with deformable tissues without the need to explicitly design policies, the robustness of learned tasks to changes in initial conditions or the environment is strongly influenced by the amount and variety of expert demonstrations provided to the system [139]. Collecting a dataset with a vast repertoire of trajectories from multiple experts and varying initial conditions is impractical and often unfeasible in clinical settings. Furthermore, with LfD, the robot's performance can only match the level of expertise demonstrated by the human, with no additional information to improve the learned behavior.

Recent developments in surgical subtask automation have demonstrated a growing interest in utilizing data-driven approaches, such as DRL [51, 347]. Several studies have employed DRL-based approaches to learn tensioning policies for multiple pinch point cutting tasks involving surgical soft tissues [114, 347]. Shin *et al.* employed an RL-based approach to learn model predictive control for tissue dynamics [50], whereas Pedram *et al.* used handcrafted features to incorporate prior knowledge in a vision-based RL approach [348].

However, a major drawback of DRL approaches is that agents only achieve robust performance after exploring a large number of possible policy options, requiring long training consisting of a significant number of attempts, which is not practical in real surgical robotic systems. Consequently, existing DRL-based works learn surgical tasks in simulated environments to enable the many trial and error attempts required to train agents in controlled settings. Simulations provide a testbed to predict unsafe or dangerous situations and prevent their execution in the real world.

Nevertheless, a key challenge when training in simulation is to minimize the reality gap, i.e., the discrepancy between the simulated and real environments, to enable successful deployment of the learnt policies in the real world, which is known as a sim-to-real approach [349]. In the context of robot-assisted MIS, this implies that the simulation should consider both the deformable properties of the anatomy and the interaction with surgical tools. This limitation is inherent in **dVRL!** (**dVRL!**), a simulation framework to train DRL agents for surgical tasks proposed by Richter *et al.* [51], which supports only rigid objects. Due to the reality gap, simulation-learnt behaviors have only been transferred to real surgical robotic systems for simplified geometries [114]. Recently, Xu *et al.* have proposed SurRoL, a simulation-based platform for DRL that can simulate deformable objects and is interfaced with the da Vinci Research Kit (dVRK) [282]. SurRoL shows promise for successful transfer of behaviors learnt in simulation to the real world.

Recent studies have suggested the combination of LfD and DRL to benefit from the strengths of both approaches and overcome their limitations [350]. In particular, demonstrations can be used to guide the exploration process during learning, reducing the time required to find an improved control policy that may differ from the demonstrated behavior. GAIL has shown to be a promising approach in endovascular manipulators but has not yet been tested in Robot-assisted MIS [351].

This chapter presents the *UnityFlexML* framework, a general and modular tool that utilizes RL methods to learn task automation in simulated surgical environments involving deformable objects. *UnityFlexML* serves as an interface between a realistic simulation of deformable anatomy, the surgical robotic system, and learning-based methods. Our main contribution is to prototype and test DRL and LfD techniques in automating the TR subtask using *UnityFlexML*. We demonstrate that the learnt policies can be successfully transferred

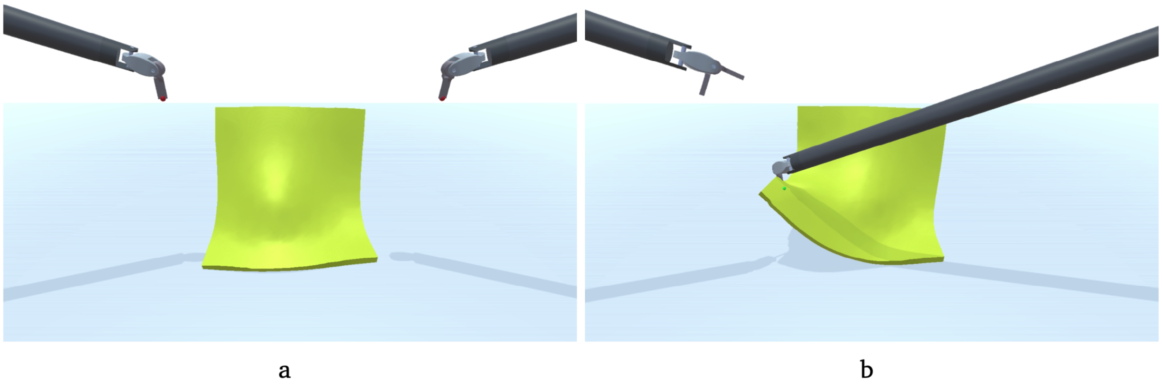


Fig. 8.1 *UnityFlexML* framework. The simulated dVSS arms interact with deformable tissues, modeled using PBD method. Example scene (a) at rest, (b) during tissue manipulation.

to the real system without additional training, thanks to the high level of realism achieved by the simulation environment.

8.2 *UnityFlexML*: a framework to learn surgical tasks in simulation

UnityFlexML[1] is a modular framework which enables the utilization of learning-based methods for task learning in a simulated surgical environment that incorporates deformable objects (refer to Fig. 8.1). The developed platform interfaces the real dVRK with a simulation of the surgical environment. We demonstrate that our simulated environment can be effectively employed to train a DRL agent to manipulate soft tissues, and the acquired policy can be successfully deployed to the dVSS, which is controlled through the dVRK [352].

8.2.1 Robot Platform

Our work involves a single dVRK unit, specifically the PSM arm. To simplify the observable state space for the RL agent in both the simulator and the real robot, we control the motion of the PSM EE in Cartesian space while keeping the EE orientation constant, as done in [51]. As long as the kinematic model for the surgical tool is loaded in simulation, the platform can accommodate any possible surgical tool. Thus, the state of the PSM EE can be described by its position $\mathbf{p}_t$ and gripper state ($g_t \in 0, 1$, open/close). Similar to [51], we normalize the PSM positions with respect to the workspace, which is defined by the PSM joint limits and obstacles in the environment. This normalization facilitates generalization of learned policies to various joint configurations. In addition, we assume that the 3D model of the anatomical environment is available, which is extracted from pre-operative images such as MRI, and this allows us to determine the position of the tumor area of interest $\mathbf{q}$.

[1] publicly available at https://gitlab.com/altairLab/unityflexml

8.2.2 Simulation Environment

Our simulation framework is built on the Unity3D engine, a game development platform that has shown potential in medical simulations [353]. The modularity of Unity makes it easy for users to customize the environment scene and to use advanced features provided by separate plugins. Specifically, our framework relies on two main Unity plugins: the Machine Learning Agents Toolkit (ML-Agents), for training intelligent agents [354], and NVIDIA FleX, for simulating soft object deformations [355].

Deformable bodies are simulated using the PBD approach, which leverages the optimized implementation provided by NVIDIA FleX. This approach has been demonstrated to accurately model soft tissue deformations in a computationally efficient and numerically stable manner [353]. These qualities are particularly important for our framework, which must enable the simulated agent to undertake multiple interactions with the environment within a short period to facilitate efficient training, while also minimizing the risk of simulation instability.

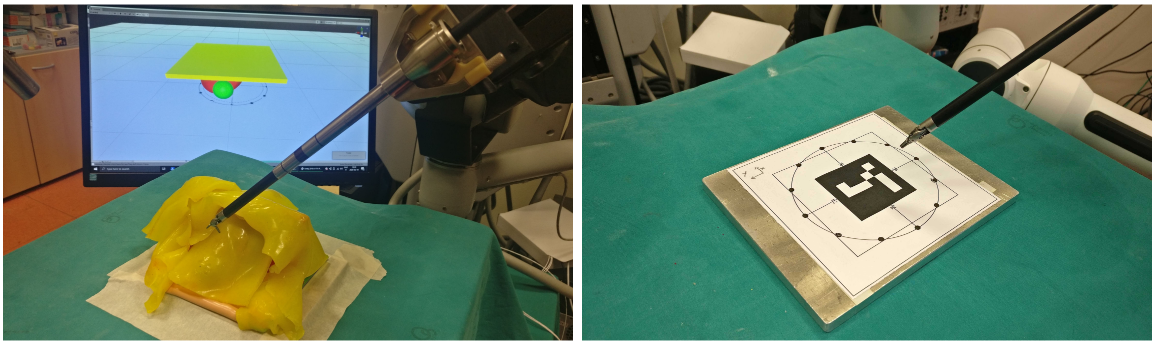
Our choice to simulate anatomical deformations using PBD rather than the finite element method is motivated by the superior performance of PBD in terms of computational efficiency and numerical stability. In the context of our framework, where the robotic agent performs numerous interactions with the environment, these aspects are essential to enable efficient training and ensure that the simulation remains stable.

To simulate the robotic part, we implemented closed form inverse kinematics of the PSM to enable Cartesian space control of the manipulator. Communication between Unity3D and Robot Operating System (ROS) is achieved using UDP-based communication, following the method described in [356]. During each simulation step, the robotic system is allowed to perform a very small motion increment, which makes the impact of the robot's dynamic behavior negligible and therefore not accounted for in the simulation [51]. Grasping of an object is modeled as an atomic event triggered when the relative distance with the EE is less than $2, mm$ in our simulated environment.

8.3 Learning tissue retraction within *UnityFlexML*

Our study focuses on learning the soft TR task using the *UnityFlexML* platform. Specifically, we consider a transanal MIS procedure for partial nephrectomy using the dVSS as our experimental setup. The TR task involves manipulating the highly deformable perirenal fat tissue to reach the adipose tissue covering the kidney, grasp it, and lift it to expose the tumor.

To achieve our goal, we adapt two approaches, namely a standard DRL approach and GAIL, to train agents in the simulated environment. Our study is motivated by the availability of synthetic phantoms for immediate testing with the real system. In the following sections,



(a) (b)

Fig. 8.2 (a) In our setup the PSM interacts with silicone fat tissue covering a kidney phantom. The simulated scene controlling PSM movements within *UnityFlexML* can be seen in the background. (b) The calibration board used to uniquely map all the components of our real experimental setup to the *UnityFlexML* environment.

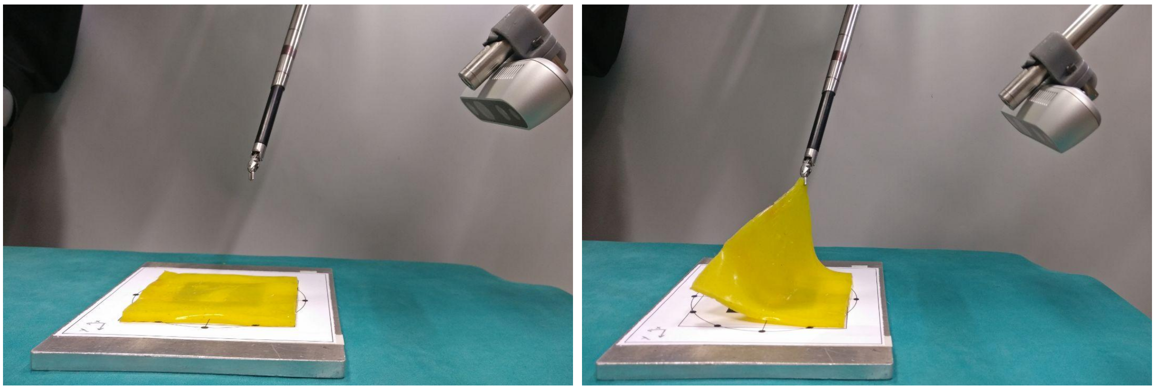
we elaborate on our experimental setup and explain how we modify the two approaches to learn the TR task using the *UnityFlexML* platform.

8.3.1 Experimental Setup

Our real experimental setup consists of a synthetic kidney phantom covered with silicone fat tissue, shown in Fig. 8.2a. The interaction between the agent and the fat tissue is limited to a 90 x 90 mm square region, which is firmly secured to the top section of the kidney. To hold the silicone patch in place, we have created a custom-designed rigid structure that allows us to determine the exact position of the fat and kidney in both the simulated and real environments. To initiate the experiment, the square region of fat tissue is allowed to fall on the kidney phantom under the influence of gravity. Our experimental scenario involves a single PSM arm that is equipped with the Large Needle Driver. All simulation experiments, including agent training and dVRK control, are conducted on a workstation that is equipped with an AMD Ryzen 3700X processor and NVIDIA TitanX GPU.

UnityFlexML environment: Simulation

The simulation environment in *UnityFlexML* is initialized with a 3D model of the kidney phantom and the position of the tumor q, obtained from segmentation of the CT of the phantom. To minimize the reality gap between the simulated and real adipose tissue, we perform an optimization procedure to determine the PBD deformation parameters that best represent the behavior of the synthetic fat tissue. A genetic algorithm is employed to optimize the PBD parameters that impact the deformable behavior of the tissue that the robot interacts with. The optimization is carried out using preliminary experiments in which a teleoperated PSM arm lifts the fat tissue from a planar configuration that is rigidly fixed



(a) (b)

Fig. 8.3 One of the experiments of the optimization process. The fat tissue is anchored to the calibration board (right side in the figure). (a) Rest condition; (b) Deformed condition. Point cloud of the deformed tissue is acquired with the depth camera shown on the right.

on one side. We define N = 5 different pinch points along the fat contour and L = 3 different levels of lifting for each pinch point. The ground truth positions of the fat tissue are obtained using an Intel RealSense D435 Depth camera (Intel Corporation, Santa Clara, USA), whose position is defined relative to a custom calibration board that ensures the alignment between the simulated and real environment (Fig. 8.2b).

The values for the cluster spacing, cluster radius, and cluster stiffness parameters that optimally control the PBD implementation of NVIDIA FleX are estimated by minimizing the error ϵ as follows:

$$\epsilon = \frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{L}\sum_{m=1}^{M}||\mathbf{x_{PBD}}(l,n) - \mathbf{x_{PCL}}(l,n)|| \tag{8.1}$$

where $||.||$ represents the Euclidean distance between the position of the M particles defining the fat in simulation $\mathbf{x_{PBD}}$, at deformation level l and pinch point n, and the closest point of the corresponding point cloud $\mathbf{x_{PCL}}$. The acquired point cloud has been decimated to bring the number of points comparable to M. The diameter of the PBD particles is set to $3mm$ (i.e., the width of our tissue sample), which allows to describe the dynamics of the fat tissue with a single layer of particles. The constraints and the range of allowed values for each parameter are set according to [353].

The optimization process yielded optimal values for the cluster spacing, radius, and stiffness parameters, which were found to be 0.127, 0.095, and 0.361 respectively. These values resulted in an average error of approximately 3 mm between the simulated and ground truth point clouds, which is in line with the dimensions of PBD particles. The optimized values were then utilized to accurately depict the deformable properties of the fat tissue in the simulation environment.

Robotic setup

The crucial first step for transferring the learned policy from the simulation scene to the real dVRK system is the precise alignment of the two environments. To achieve this, we reach several points on the calibration board displayed in Fig. 8.2b to map the poses of the PSM in a common reference space. The accurate registration of the two environments is of utmost importance for our application, as all the movements of the da Vinci arm in the real system are directly controlled by the simulated environment. Additionally, since there is no visual feedback used in these preliminary experiments, grasping events are triggered in simulation upon detecting collision events between the end-effector and the fat, and the corresponding action is transmitted to the real system. Therefore, it is vital to ensure accurate registration to avoid any inconsistencies between the two environments. The mean positioning error of the PSM arm is 1.7 mm.

8.3.2 Learning methods

In this study, we represent the agent using the EE of the da Vinci PSM, which interacts with the surrounding anatomical environment. The initial state of the environment is assumed to be known from pre-operative data. Our objective is to move the PSM arm from a pre-defined initial position $\mathbf{p}_0$ to a position close to the tumor $\mathbf{q}$, grasp the fat and lift it to a pre-defined final position $\mathbf{p}_T$ in order to expose the tumor. To ensure that the learned motion primitives are robust to different initial configurations, the EE starts from a different position $\mathbf{p}_0$ after each episode (2500 timesteps) during training. The position $\mathbf{p}_T$, on the other hand, remains fixed throughout the training experiments. We define the state space using kinematics information to describe the current robot state and environment at time t:

The state and action space of the environment is:

$$\begin{aligned} S_t &= [\mathbf{p}_t, \mathbf{q}, \mathbf{p}_T, ||\mathbf{p}_t - \mathbf{q}||, ||\mathbf{p}_t - \mathbf{p}_T||] \\ A_t &= [\Delta_t, g_t] \end{aligned} \tag{8.2}$$

where $||.||$ is the Euclidean distance. $\Delta_{t,i} = 0.5\alpha$, $\alpha \in \{0, -1, +1\}$ tells the agent if it has to remain still, move backward or forward by $0.5\,mm$ in the i_{th} spatial dimension, while $g_t \in \{0, 1\}$ represents the gripper state (open/close).

The feasibility of using UnityFlexML to learn a surgical task is evaluated using two possible strategies: a standard DRL approach and GAIL.

8.3.3 DRL setting

We use a consolidated DRL algorithm called PPO (described in Sec. 3.1.3) provided by Unity3D ML-Agents plugin [357]. The architecture of the actor-critic networks of the PPO agent used is shown in Fig. 8.4.

For the training phase, we design a reward function which changes depending on the current gripper state:

$$r(s_t) = \begin{cases} ||\mathbf{p}_t - \mathbf{q}|| * k - 0.5, & \text{if } g_t = 0 \\ ||\mathbf{p}_t - \mathbf{p}_T|| * k, & \text{if } g_t = 1 \end{cases} \tag{8.3}$$

where k is a normalization factor which depends on the volume in which PSM can move. When the gripper is open, the reward encourages the PSM to move towards the tumor. On the other hand, when the EE has grasped the tissue, it is pushed towards the target position. During the training phase, rewards are accumulated at each episode, which ends after 2500 steps. As this approach is a pure DRL method, it is entirely trained in simulation. Henceforth, we refer to this setting as "PPO".

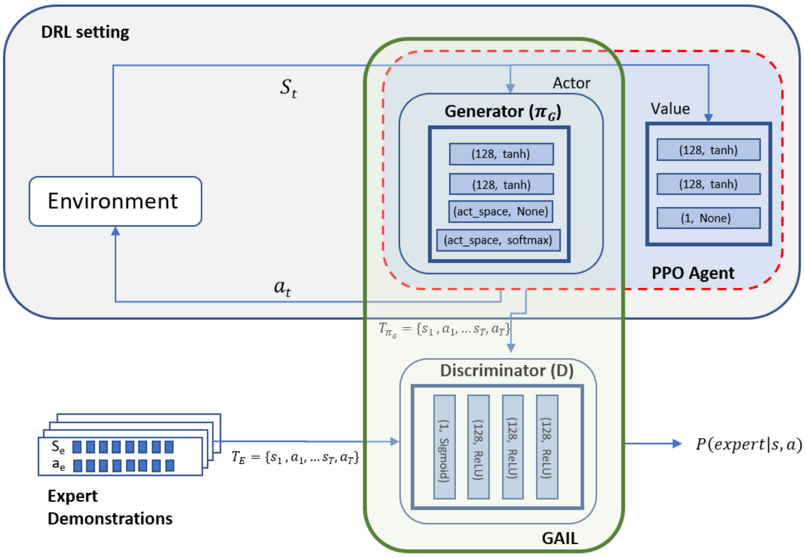


Fig. 8.4 Network architecture of GAIL and PPO. PPO consists of a policy (actor) and Value network. The policy network acts as Generator for GAIL. Generated trajectories and expert trajectories are passed to the Discriminator. Discriminator learns a probability function which classifies the generator trajectory as expert or non-expert. The network layer details are depicted inside each box in the format (hidden units, activation) respectively.

8.3.4 GAIL setting

The second approach considered in this study is based on the learning paradigm of GAIL, which uses a policy generator that builds on PPO. The architecture of the network used for this approach is illustrated in Fig. 8.4. The loss function employed in this setting is a linear combination of DRL and GAIL losses, with $\alpha L_{DRL} + \beta L_{GAIL}$ representing the weighting factors for the two loss functions. Our initial investigation into hyper-parameter tuning revealed that the best performance was achieved with $\alpha = 0.2$ and $\beta = 0.8$, as other values resulted in slower convergence.

Training a GAIL agent requires the collection of trajectory demonstrations. In this study, task demonstrations were obtained on the real dVRK and transferred to the *UnityFlexML* framework. The acquired trajectories were repetitive fat lifting tasks performed by an expert user. As the expert user was aware of the final objective of exposing the tumor, the grasp position was near the tumor for all the demonstrations. Additionally, the expert user was instructed to vary the trajectories by starting each demonstration from a different initial position above the fat surface.

Acquisition of task demonstrations from the real environment leverages the communication pipeline provided by *UnityFlexML* (Fig. 8.5). Registration between the simulated and real environment is guaranteed following the same registration process described in [52]. The joint values are sent to *UnityFlexML* through UDP sockets and the desired configuration is reached with direct kinematics. Each recorded demonstration consists of the set of kinematic observations that define the state space (Sec. 8.3.2) and the corresponding actions at each timestep. An important aspect of this implementation is the challenge associated when we reset each episode. In the simulation, as soon as the target position is reached, the grasp is released and the episode resets. The position of the EE is then immediately teleported to the next initial point. Such an instantaneous reset strategy is difficult to model in the real robotic system. Hence during the recording of expert demonstration, a delay of some timesteps has been added between the moment when the grasp is released and the beginning of the next episode, to allow repositioning. We make use of 35 continuous episodes recordings. Although our simulation framework supports demonstration recordings using a keyboard or a joystick, the established communication pipeline between dVRK and *UnityFlexML* is crucial since it helps to acquire demonstrations directly with the real robotic system, thus without deviating from the surgical workflow.

The acquisition of task demonstrations from the real environment is facilitated by the communication pipeline provided by *UnityFlexML* (as depicted in Fig. 8.5). To ensure registration between the simulated and real environment, the same registration process as described in [52] is followed. The joint values are sent to *UnityFlexML* via UDP sockets and the desired configuration is achieved using direct kinematics. Each recorded demonstration comprises the set of kinematic observations that define the state space (as discussed in Sec. 8.3.2) and the corresponding actions taken at each timestep.

An important challenge in this implementation is associated with resetting each episode. In the simulation, the reset strategy involves an immediate release of the grasp and teleportation of the EE to the next initial point once the target position is reached, effectively resetting the episode. However, replicating this strategy in the real robotic system is challenging. To address this, a delay of several timesteps has been added during the recording of expert demonstrations between the moment the grasp is released and the start of the next episode, allowing for necessary repositioning. We have recorded 35 continuous episodes during expert demonstrations using the established communication pipeline between the dVRK and

UnityFlexML. Although our simulation framework supports demonstration recordings using a keyboard or a joystick, acquiring demonstrations directly with the real robotic system via the established communication pipeline is essential as it helps to obtain demonstrations without deviating from the surgical workflow.

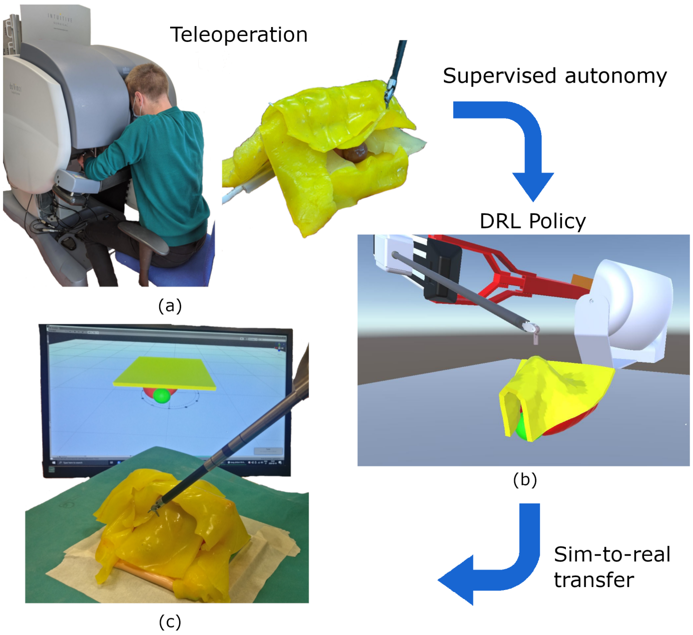


Fig. 8.5 The proposed methodology of LfD for the tissue retraction surgical gesture. (a) Expert demonstrations are performed and recorded using the dVRK console (b) Robotic agent is trained within a simulated environment. (c) The learnt policy is translated to the real robotic system.

8.3.5 Evaluation metrics

Evaluation metrics have been defined to assess the suitability of the presented framework for learning surgical tasks. Specifically, the performance of the considered methods to learn the TR task is tested when training within *UnityFlexML*. The high level of realism of the simulated environment created within *UnityFlexML* not only allows for training the methods with a sim-to-real approach but also provides a platform for testing the presented methods in realistic settings. As a consequence, the learnt behavior is tested both in a simulated environment, provided by *UnityFlexML*, and in the real one, in a sim-to-real fashion. To evaluate the performance of the algorithms, two criteria are considered: sample efficiency and optimality of the accomplished task.

Sample efficiency is defined as the amount of experience an algorithm needs to learn a behavior by interacting with the environment. It is estimated as the number of time steps required by each algorithm to reach high reward values. On the other hand, optimality of the learnt behavior represents the ability of each method to make the tumor visible upon task completion, and is assessed using a TE metrics. To compute TE, the image captured by an endoscope positioned in front of the kidney is considered for both the simulated and real setup. A circular region of interest around the tumor is selected, and the visible portion of the tumor is extracted by applying a mask with HSV bounds matching tumor color (Fig. 8.9). The TE is then computed as the percentage of tumor pixels that are visible within the region of interest, normalized in the range [0, 1].

8.4 Autonomous tissue retraction in simulation

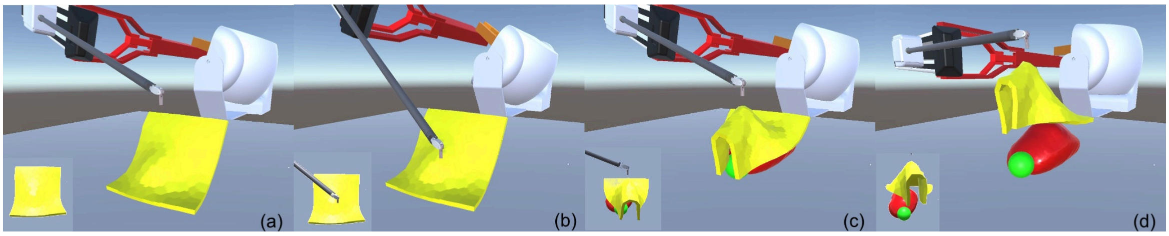


Fig. 8.6 Sequence of action frames for task completion in simulation: (a) approach, (b) grasp, (c) retract, (d) expose. Perspective of the simulated camera is overlaid on the bottom left of each simulator frame.

The performance of the two presented methods in achieving the TR task is evaluated in simulation after training within *UnityFlexML* (Fig. 8.6). The evaluation experiment involves the trained agents performing the TR task starting from 49 different positions. These positions are uniformly sampled on a 7x7 regular grid above the portion of the fat tissue. This experiment is designed to assess the robustness of the learned behavior of the agents to different starting positions $\mathbf{p}_0$ of the EE. The TE metrics is used to evaluate the agents' performance each time the EE reaches $\mathbf{p}_t$.

8.4.1 Results and Discussion

The results and discussion of the experiment are presented in this section. The learning curves of the two considered learning configurations, i.e., GAIL and PPO, are shown in Fig. 8.7. Both methods aim to maximize the cumulative reward, but they exhibit different learning patterns. The learning curve of GAIL is smooth and monotonous, gradually increasing towards high-reward values. In contrast, the curve of PPO shows a modular reward trend, requiring 2.5 million steps to learn the approach behavior and interaction with the fat, and another 1 million steps to learn the retract behavior.

It is observed that GAIL is more sample efficient than PPO since it requires fewer steps to learn the task. This experiment confirms that incorporating human demonstrations can make the learning process more efficient than the baseline PPO, highlighting the benefits of incorporating human knowledge into the learning process.

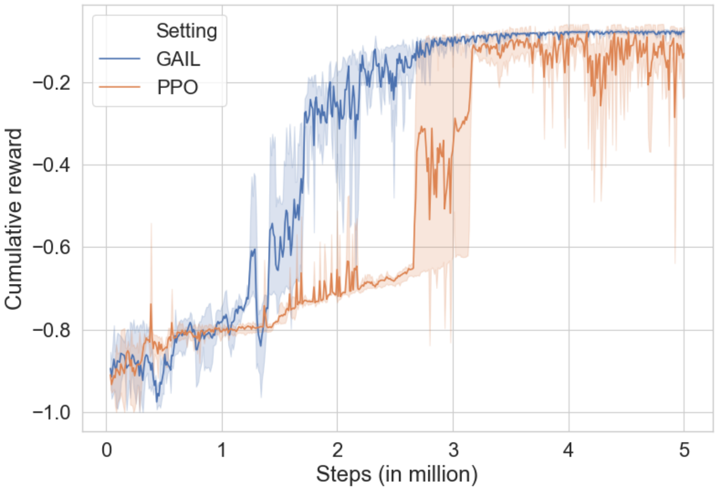


Fig. 8.7 The obtained learning curve for GAIL and PPO. Cumulative reward is normalized in the range $[-1, 0]$. The shaded area spans the range of values obtained when training the agent starting from three different initialization seeds.

The plot in Fig. 8.8 shows the results of the simulation experiment, where the TE from the simulated camera was analyzed depending on the starting position of the PSM arm above and outside the boundary of the fat tissue. The agents trained with both PPO and GAIL were able to grasp the tissue and partially expose the tumor when starting from the distal part of the tissue, which is the part farthest from the fixed region. However, it was observed that PPO achieved little or no tumor exposure when the starting EE position was close to the fat attachment (Fig. 8.8a), even though the agent has learned how to perform the task correctly (Fig. 8.7). It seems that, when $\mathbf{p}_0$ is initialized close to the fixed fat region, the agent is not able to move towards a reasonable grasping point, thus causing the tumor not to be exposed. This suggests that the reward function used in the experiment was suboptimal for the task, as it encouraged the agent to approach the known position of the tumor, but abruptly changed as soon as the EE was in contact with the tissue, regardless of its current grasping position. Therefore, the agent might end up grasping at a suboptimal location, which is not ideal for exposing the tumor. Manually tuning the reward function to encode complex task objectives such as tumor exposure might be challenging, especially relying on kinematic data alone. However, including a TE-dependent term into the reward function could potentially improve the learned behavior towards the task objective. Preliminary evaluation with a reward function including a TE-dependent term did not show significant improvements in the results, possibly due to the sparse reward scenario where TE is always zero before grasping. Future works will investigate this further.

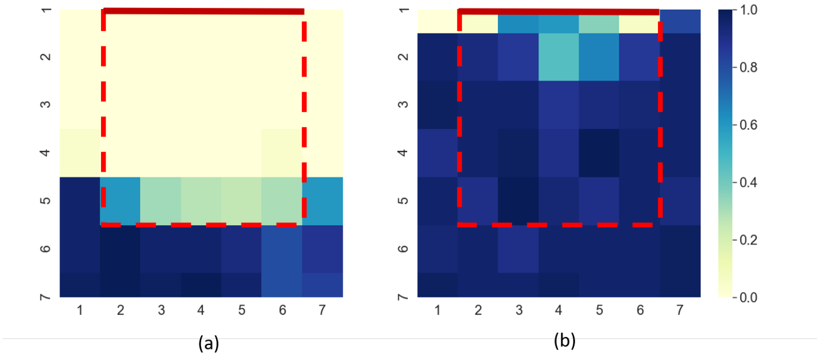


Fig. 8.8 Simulation experiments: TE from the camera at different initial positions, of the PSM for (a) PPO, (b) GAIL. The color of each subregion is related to the percentage of visible tumor area when $\mathbf{p}_0$ belongs to that subregion. The fat boundary from the top view is depicted in red dashed lines whereas the fat attachment is shown in the solid red line

In contrast, the incorporation of human demonstrations using GAIL results in a learnt behavior that enables the successful exposure of the tumor regardless of the initial position of the PSM arm (Fig. 8.8b). It is important to note that the strategy employed by the human demonstrator involves moving and grasping towards points in close proximity to the tumor, with the objective of maximizing exposure. The primary difference between the behavior learnt by PPO and GAIL lies in the selection of the grasping point at different starting positions. Specifically, when the starting position is above the attached area, GAIL grasps closer to the tumor, resulting in a higher TE, as it learns to imitate the human operator who moves towards the most appropriate points to maximize exposure.

8.5 Sim-to-real autonomous tissue retraction

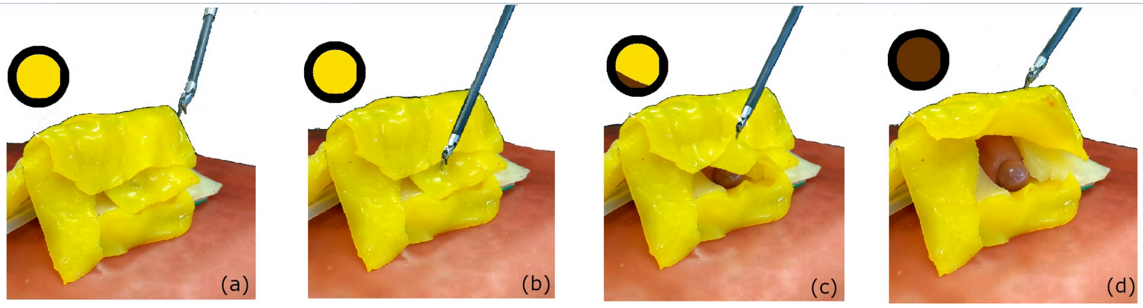


Fig. 8.9 Sequence of action frames for task completion in real world setup, with the circular mask used to compute TE metrics. (a) approach (TE=0%), (b) grasp (TE=0%), (c) retract (TE=∼15%), (d) tumor exposure (TE=100%). The real camera is placed in front of the phantom, in the same position as in the simulation (which does not correspond to the viewpoint of these pictures).

The behavior learnt with the considered approaches using *UnityFlexML* is transferred to the real robotic platform. Two main factors affect the ability to achieve this transfer. Firstly, the level of realism of the simulated environment used for training plays a critical role, as the agent can only learn the correct task if the reality gap is minimized. Secondly, the accuracy of alignment between the simulated and real environments plays a crucial role, as all movements of the dVRK arm in the real system are controlled via the simulated robot, including the grasping action.

8.5.1 Results and Discussion

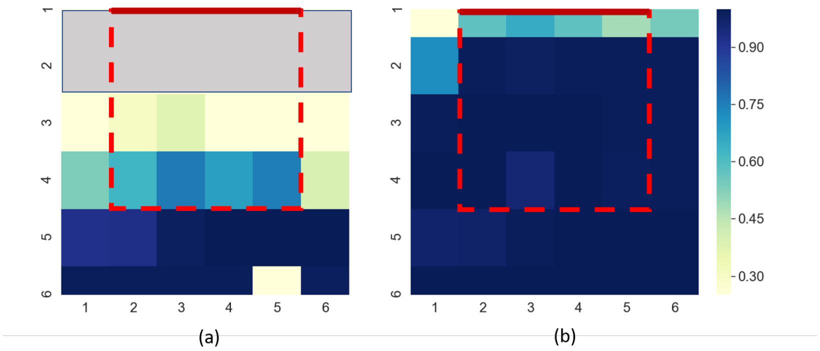


Fig. 8.10 Real grasp experiments: TE from the camera when starting from different initial positions of the EE, using (a) PPO (b) GAIL. The portion of fat tissue which is not considered for the experiments is colored in gray.

We have successfully replicated the learned behavior from the simulated environment to the real robotic platform without any inconsistencies. The end-effector of the dVRK arm was able to contact the fat tissue and reach the target point from all different initial positions. The TE percentage starting from various points above the real fat tissue is depicted in Fig. 8.10. To avoid tissue tearing that may occur during grasping too close to the attached area, we did not attempt starting positions near the attachment when testing the behavior learned with PPO, which is represented as the unattempted gray region in Fig. 8.10a.

When considering the average TE over all trials from different starting points, PPO achieves an average TE of 0.38 while GAIL obtains an average TE of 0.90. Comparing the results obtained for GAIL and PPO, it is evident that GAIL is able to reach higher overall exposure and is more robust to changes in the initial PSM position. GAIL is also capable of achieving tumor exposure from starting points that were unattempted for PPO, indicating a more optimal learned trajectory and superior overall performance. This observation suggests that the initial PSM position significantly affects the performance of PPO, while GAIL is capable of achieving optimal performance regardless of the starting position, consistent with the results obtained in the simulated experiments. In summary, our results demonstrate that using demonstrations is a robust and superior approach compared to PPO in both simulated and real-world experiments.

8.6 Conclusion

In this chapter, we have developed and implemented *UnityFlexML*, a modular framework that enables simulation of deformable objects. Through *UnityFlexML*, standard DRL approaches can be trained in simulation and the learnt behavior can be translated to the real robotic system (specifically, dVRK). Moreover, expert users can execute tasks on the real system and these executions can be used for DRL training in simulation.

Furthermore, we have presented an LfD methodology based on GAIL for automating TR. This approach can learn generalized, human-like trajectories in a sample-efficient manner by utilizing a well-established DRL architecture. Our experiments in simulation and the real environment demonstrate that while both baseline DRL methods and GAIL can complete the task, the latter can reduce the required number of steps and produce near-human trajectories, thus improving the learning process. The policies learnt by both methods exhibit robust performance when deployed on the dVRK.

However, there are some limitations to this study. The underlying assumption is based on knowing the target positional coordinates (such as tumor position) pre-operatively. In reality, the TR surgical gesture may need to be carried out as an exploratory subtask without a known target. Hence, our future work will focus on utilizing visual information to estimate the kinematic coordinates of various image features, as described in [58, 358]. Furthermore, further experimentation is needed to assess the impact of the quality and number of demonstrations required to learn optimal behavior for different surgical gestures, involving experts with varying levels of expertise. Additionally, it is important to consider the safety issues that may arise due to free exploration of the state space, which may result in dangerous movements. Therefore, in future work, we plan to incorporate safety constraints through a Safe-RL technique [359].

Contributions of this chapter

1. Introduction of *UnityFlexML*, an open-source modular framework that provides an interface among a realistic simulation environment supporting deformable objects, the surgical robotic system and learning-based methods. *UnityFlexML* is available at https://gitlab.com/altairLab/unityflexml.

2. The proposed framework has the required features to allow learning a surgical task (i.e., tissue retraction) both using a standard DRL method and a strategy combining DRL with LfD.

3. The learnt policy translates directly to the surgical robotic system thanks to the da Vinci Research Kit (dVRK), without further training.

Publications linked to this chapter

1. Eleonora Tagliabue*, Ameya Pore*, Diego Dall'Alba, Enrico Magnabosco, Marco Piccinelli, and Paolo Fiorini. "Soft tissue simulation environment to learn manipulation tasks in autonomous robotic surgery." In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3261-3266. IEEE, 2020.
   * Equal contribution

2. Ameya Pore, Eleonora Tagliabue, Marco Piccinelli, Diego Dall'Alba, Alicia Casals, and Paolo Fiorini. "Learning from demonstrations for autonomous soft-tissue retraction." In 2021 International Symposium on Medical Robotics (ISMR), pp. 1-7. IEEE, 2021.

Chapter 9

Safety in tissue manipulation using formal verification

9.1 Introduction

In Chapter 8, a comprehensive examination of recent advancements in surgical subtask automation was undertaken, and it was noted that a growing interest has emerged in utilizing data-driven methodologies, such as DRL [51, 347]. Despite their promising results, the training of DRL methods is premised on the efficient exploration of state space and does not explicitly account for the risk associated with actions [265]. DRL algorithms find an optimal policy by maximizing long-term rewards, but this does not address the potential for infrequent negative rewards, which may correspond to high-risk actions.

To address this challenge, DRL models are typically trained in virtual environments. This approach is particularly useful for robot-assisted MIS where strict ethical, legal, and economic constraints require the validation of automation methods in a simulated environment before implementation in real-world scenarios. Recent works have proposed surgical simulation environments suitable for training DRL algorithms [51, 52]. Nevertheless, concerns regarding the safety of DRL methods have limited their deployment in a clinical setting.

The guarantee of a provable behavior using DRL remains an open problem, and its resolution is crucial for building trustworthy solutions for universal applications [360]. While DRL methods have shown promising results in surgical subtask automation, the lack of consideration of risk and provable behavior remains a challenge that must be addressed to ensure the safe and reliable deployment of these methods in a clinical setting.

The concept of safety, and its counterpart, risk, is closely tied to the inherent uncertainty and stochasticity of the environment. Different perspectives have resulted in various definitions of safety, as reviewed in [361]. In the context of this chapter, we define safety as a condition that is unlikely to result in harm or injury. Humans must classify environmental states as *safe* and *unsafe*, and agents are deemed *safe* if they never encounter *unsafe* states.

To address safety in DRL, a recent research direction involves incorporating auxiliary objectives into the training process to enhance safety. Multi-objective RL seeks to optimize an additional cost function that measures safety [362], for example, by counting the number of collisions. However, the challenge of explicitly learning behavior over multiple objectives can result in either an average policy [363] or scalability issues [362]. Similarly, Constrained RL [364] introduces safety constraints during the training phase by limiting the accumulation of the cost function. However, these approaches lead to a significant trade-off in functional performance as the constraints severely restrict the exploration process, affecting the learned behavior.

A more intuitive approach to address safety in well-defined tasks is through reward shaping [265]. The idea is to use domain knowledge to design proxy reward functions that lead the trained policy to perform desired safe behaviors, which can be naturally incorporated into well-defined training procedures such as the one considered in our work. In conclusion, incorporating safety into DRL remains a challenge, but reward shaping offers a promising approach for well-defined tasks.

We described in Chapter 6 that the utilization of DNN as the underlying mechanism for DRL decision making can result in unforeseeable behavior if the network encounters input data that falls outside of the training regime. Thus, ensuring that the DNN never produces decisions that lead to safety violations is crucial. Such validation requires estimating violations without executing the network, i.e. without performing the actions in a DRL setup. Running the network over many experiments and counting the unsafe configurations can be time-consuming and can only give an empirical evaluation without any guarantee of safety [360].

One of the earliest approaches to evaluate the robust nature of DNN was ReluPlex [305], which aimed to find the largest neighborhood in the feature space that guaranteed that no point within that area would change the classifier's decision (i.e., small perturbations in the input does not change the network decision). However, such verification is NP-complete and does not scale well in large input spaces [365]. Another approach, formal analysis using interval algebra [366], has been adopted to verify handcrafted safety properties [307]. FastLin [367] utilized the linear approximation of ReLU units to offer an efficient and scalable algorithm, while Neurify [368] relied on symbolic interval analysis to provide a strict estimation of output bounds within a subset of the input space. However, these methods cannot be easily adapted to DRL scenarios, where a network encodes a sequential decision-making problem and lacks metrics to evaluate safety.

For these reasons, we have adapted standard approaches and formulated a FV tool that enables us to mathematically guarantee the safety of the learned behaviors with respect to pre-defined safety rules, referred to as properties. Furthermore, we have defined a metrics, called the violation rate, which allows for the evaluation of how often a trained DRL model (under small adversarial perturbations) will violate these properties.

In summary, we present a Safe-DRL framework for the automation of the TR surgical subtask. The safety issue in TR is defined as a set of properties that outline the safe operating parameters, ensuring that the PSM does not collide with surrounding anatomy. To assess safety, we employ a FV analysis that quantifies the likelihood of unsafe configurations relative to the established safety rules. The experimental scene consists of a virtual environment (developed in Chapter 8) for a robot-assisted MIS procedure that extensively requires manipulation and TR of fat tissue that covers the kidney to expose the region of interest (see Fig. 9.1a). One of the challenges in automating TR is to accommodate the variable and dynamic properties of the deformable tissue while preserving the surrounding structures [358]. Our contribution addresses this challenge by introducing a framework for safely automating surgical subtasks using DRL methods and by providing a tool for FV to evaluate the compliance with safety properties.

9.2 Safe-DRL for TR

Our aim in this study is to successfully perform the task of TR, which involves exposing the tumor while avoiding interaction with the surrounding organs and tissues. In order to do so, we consider a surgical scenario that involves the use of a dVSS robotic PSM and several organs, including a kidney covered by a layer of perirenal fat tissue (as illustrated in Fig. 9.1a). We utilize the *UnityFlexML* framework to simulate the behavior of the deformable fat tissue, as described in Chapter 8.

As depicted in Fig. 9.1b, *UnityFlexML* allows for the integration of mesh colliders in our 3D organ models, enabling automatic detection of collisions between the PSM and anatomical organs. This, in turn, enables us to shape the rewards based on collision information. Specifically, a collision is defined as an atomic event that occurs when the bounds of two or more meshes intersect with each other. By appropriately shaping the colliders for the various components in our training scenario, we are able to detect and avoid undesired collisions.

9.2.1 Observation and Action Space

The DRL algorithm utilized in this work is embodied in the PSM EE. This is in accordance with the approach described in Chapter 8, where it is assumed that anatomical information, such as the location of organs and tumors, is obtained from pre-operative data. The task of TR involves moving the PSM from an initial position p_0 to the desired position p_{tumour} close to the tumor, and lifting the fat tissue to reach the target location p_{target}, thereby exposing the tumor. It is important to note that during training, the initial position of the PSM p_0 is randomly determined at the start of each episode. The state and action spaces of the environment are defined as follows:

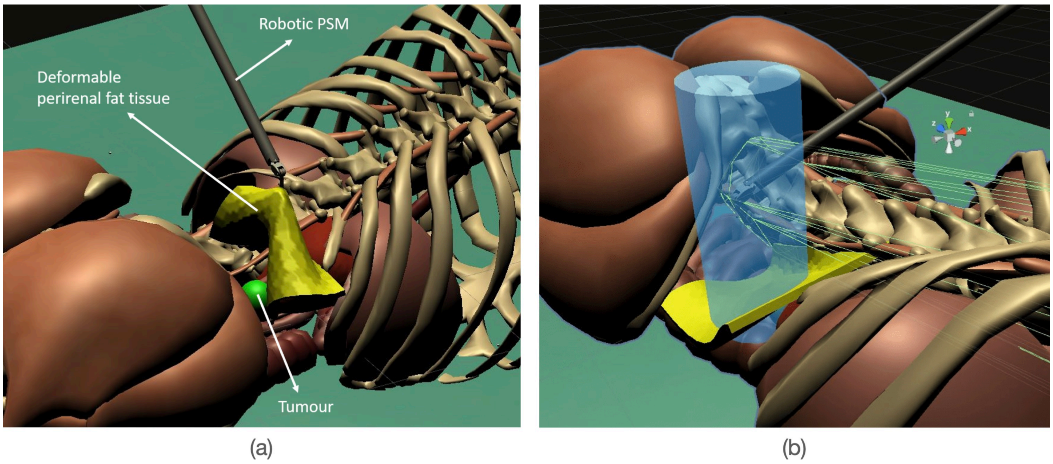


Fig. 9.1 Virtual scene used to simulate the TR task. (a) The yellow tissue represents the renal adipose tissue that needs to be retracted to expose the tumor (green sphere) embedded in the underlying kidney (not visible in the picture). (b) Explanatory overview of the safe EE workspace (light blue cylinder) and the mesh colliders (green lines) for the spinal column.

$$\begin{aligned} S_t &= [g_t, p_t, p_i, \|p_t - p_i\|] \\ A_t &= [\Delta_{t,j}] \end{aligned} \tag{9.1}$$

where $g_t \in \{0,1\}$ is the gripper state (open or close), p_t is the position of the EE, p_i is either p_{tumor} in the first part of the trajectory (i.e., $g_t = 0$) or p_{target} in the lifting part (i.e., $g_t = 1$), and $\|.\|$ is the Euclidean distance between the EE current position and the current target. In the action space, $\Delta_{t,j} = 0.5\alpha$ (with $\alpha \in \{0,-1,+1\}$ controls the EE to move backward or forward by $0.5mm$ in the j_{th} spacial dimension, or remain still.

9.2.2 Reward Shaping

The state of the gripper, g_t, plays a crucial role in determining the goal in the agent's observation space. As a result, the reward function is designed based on the value of g_t and the proximity to the goal. The mesh collision system of *UnityFlexML* is utilized to impose a penalty term, c, to the reward when the EE moves outside of the designated workspace (depicted as a light blue cylinder in Fig. 9.1b) or the PSM arm comes into contact with any of the organs.

$$r(s_t) = \begin{cases} -(\|p_t - p_{tumour}\| \cdot k - 0.5) - c, \text{ if } g_t = 0 \\ -\|p_t - p_{target}\| \cdot k - c, \text{ if } g_t = 1 \end{cases} \tag{9.2}$$

where k is a normalization factor, and c is a constant penalty set to 1 in case of collisions. Note that the scalar quantity of -0.5 is added to restrict the reward in the range [-1.0, -0.5]

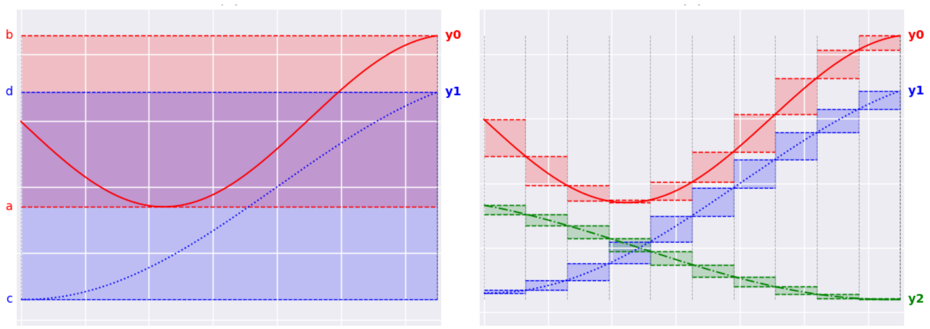


Fig. 9.2 Explanatory output analysis of (left) decision-making problem with two outputs and one subdivision, and (right) output analysis with three outputs and multiple subdivisions

before grasping and [-0.5,0] after grasping. The reward function encourages the PSM to move towards the tumor when the gripper is open and towards the target position when the gripper is closed.

9.2.3 Training Algorithm

To evaluate the performance of various DRL algorithms, we interfaced the *UnityFlexML* environment with an external Python-based DRL software module. Among the algorithms considered, including Twin Delayed DDPG (TD3) [369], Soft Actor-Critic (SAC) [370], PPO [371] and others discussed in Chapter 3, we chose PPO as it demonstrated the best overall returns in terms of hyperparameter tuning and training time. Our main objective was not to obtain the best performance, but rather to demonstrate the impact of safety constraints in DRL training. In particular, we used the ϵ-clipped implementation with $\epsilon = 0.2$ as recommended in [371].

9.2.4 Formal Analysis

Our framework for FV aims to determine the compliance of a set of properties by either confirming satisfaction or providing instances that contradict the properties. Our approach involves formalizing the safety properties using the methodology presented by Liu *et al.* [360]. This methodology expresses the relationship between inputs and outputs as follows:

$$\Theta : x_0 \in [a_0, b_0] \wedge \ldots \wedge x_n \in [a_n, b_n] \Rightarrow y_j \in [c, d] \tag{9.3}$$

where $x_k \in X$ (i.e., input space), with $k \in [0, n]$, where n denotes the size of input states (i.e. dimension of X) and y_j is a generic output of the network. Here, $a_k, b_k, c, d \in \mathbb{R}$ represents the input and output bounds, respectively.

The property formulation is intended to confirm if the network's output falls within a specified range. However, in the context of DRL, the network represents a decision-making issue where each output node signifies the value or likelihood of a specific action. The agent opts for the action with the highest probability or value with some degree of randomness. Hence, we reformulate Proposition. 9.3 to examine if one of the output values is lower than the others as follows:

$$\Theta : x_0 \in [a_0, b_0] \wedge ... \wedge x_n \in [a_n, b_n] \Rightarrow y_j > y_i \tag{9.4}$$

To verify the property, we rely on the Moore's comparison rules for intervals [366, 372]. In particular, assuming $y_i = [a, b]$ $(a, b \in \{a_k\}, \{b_k\})$ and $y_j = [c, d]$, we obtain the proposition:

$$b < c \Rightarrow y_i < y_j \tag{9.5}$$

To obtain an estimation of the output given an input interval, we utilize a layer-by-layer propagation approach [1]. However, even if the estimated bounds perfectly match the real maximum and minimum values that the output nodes could assume, as shown in Fig. 9.2 (left), we cannot formally guarantee that the property is respected. For example, in the figure, y_1 is lower than y_0 throughout the entire input domain, but due to the estimated bound limits of $y_0 = [a, b]$ and $y_1 = [c, d]$, we cannot formally determine whether the decision-making property is proved or denied using Proposition 9.5, because $d \nless a$. To summarize, FV based on Proposition 9.5 only considers the estimated bound limits to verify a property and therefore, typically fails at directly verifying properties on large input domains.

To overcome this challenge, we propose dividing the input domain of the property into a set of sub-intervals (*subarea*) and analyzing them independently. Fig. 9.2 (right) illustrates this process, where the sub-intervals allow for a better estimation of the output function's shapes and bounds, making it easier to apply the Moore rules for the interval comparison. It is possible for a situation similar to Fig. 9.2 (left) to occur in a certain *subarea.* To address this, we can recursively iterate the process until $d < a$ (property verified) or $c > b$ (property violated) for that particular *subarea.* In the right Fig. 9.2, the property $y_1 < y_0$ is proven for the entire domain, while the property $y_1 < y_2$ is clearly violated in the second half of the input domain.

This formulation represents one of the first attempts to apply formal analysis techniques to a RL problem.

9.2.5 Violation Rate

In this section, we introduce a novel metrics, derived from our FV approach, to assess the safety of a trained model with regards to a specified set of safety properties. Conventional verification algorithms have the drawback of only providing a binary outcome, either "yes" if the property holds across the input domain or "no" if the property is violated in at least one

[1] Project implementation: https://github.com/Ameyapores/SafeRLSurgery

point. Our proposal involves computing the percentage of the input domain that violates the desired properties to evaluate the model's safety. The approach entails keeping track of the sub-interval size at each iteration of our method that violates the properties. Ultimately, we obtain a *violation rate* which is the size of the violating sub-intervals normalized by the initial size of the input domain. The violation rate serves as an upper bound on the probability of violating the safety properties.

9.3 Safety Evaluation

The task of TR is split into two stages: approaching the tumor and retracting the fat tissue once it has been grasped. Our aim is to demonstrate that the overall safety of the surgical procedure can be increased by utilizing safety criteria, such as the collision penalty. To achieve this, we have established safe workspaces for both subtasks, where there is no collision between the PSM and the surrounding tissue within the workspace. Fig. 9.1b depicts the safe workspace for the approach phase. This reflects the surgical scenario in which we avoid collisions with hard anatomical structures, such as the ribs and spinal column, that can result in serious consequences, whilst ignoring the collision with soft tissues near the area of interest.

For each workspace, we have defined safety properties, including an upper and lower bound, so that a configuration of the PSM that satisfies the property is considered safe. In the approach phase, we have established properties for each direction in the Cartesian space, with Θ_{1R} and Θ_{1L} representing the left and right constraints in the x-direction, and Θ_{2R}, Θ_{3R}, and Θ_{3L} representing the constraints in the y and z directions, respectively. It is important to note that, for the approach phase, we do not impose an upper limit on the y-axis Θ_{2L} as there are no obstacles in that direction. Similar properties, $\Theta_{4R} - \Theta_{6L}$, have been defined for the retract phase. A detailed description of all proposed properties can be found in Table 9.2.

We have trained the PPO algorithm with these safety properties, referred to as Safe-PPO, by penalizing the agent if it violates these properties, i.e. by moving outside the safe workspace, as described in Sec. 9.2.2. Our experiments include a comparison between the performance of Safe-PPO in achieving high rewards and that of PPO that does not take into account safety constraints, referred to as Unsafe-PPO. We also report the violation rate of all properties for both Safe-PPO and Unsafe-PPO using formal analysis.

Additionally, to determine the impact of each considered property on the overall behavior, we have conducted an ablation study, in which we have trained PPO using a subset of properties and calculated the violation rate. Table 9.1 provides details of the selected policies that have been trained considering various properties.

Subsequently, we assess the feasibility of determining beforehand the input states that may result in unsafe configurations using the trained Safe-PPO model. Our FV tool partitions the input domain into smaller intervals and then iteratively refines the division using various

Table 9.1 Considered policies used in the ablation study.

Brains	Properties used for training
Safe-PPO	All properties ($\Theta_{1R} - \Theta_{6L}$)
Unsafe-PPO	No properties
Primitive Safe-PPO	Safe-PPO in early stages (after 400 epochs) of the training ($\Theta_{1R} - \Theta_{6L}$))
Policy4	First set of properties ($\Theta_{1R}, \Theta_{1L}, \Theta_{2R}$)
Policy5	Second block of properties ($\Theta_{3R}, \Theta_{3L}, \Theta_{4R}$)
Policy6	Last set of properties ($\Theta_{4L}, \Theta_{5R}, \Theta_{5L}, \Theta_{6R}, \Theta_{6L}$)

heuristics until it can prove or disprove the violation criteria for each interval, as detailed in Sec. 9.2.4. This allows us to identify all state values of the considered inputs that could result in violations for the DRL policy. To determine whether a standard execution using Safe-PPO encounters states that cause violations, we analyze the inputs over 1000 episodes and visualize the state distribution.

Furthermore, we evaluate the model's capability in exposing the tumor using the TE metric. TE calculates the normalized percentage of the tumor surface that is visible from a camera placed in front of the area of interest. The safe workspace is divided into a 5x5 grid aligned with the x-z plane, as shown in Figure 9.1b. The EE is positioned at each point in the grid and the number of pixels of the tumor is recorded through the camera. This assessment enables us to examine the effect of the safety constraints on tumor exposure as the initial position of the EE is varied.

9.3.1 Results and Discussion

In the training phase, both Safe-PPO and Unsafe-PPO policies completed the task in approximately 800 epochs, with each epoch consisting of 2000 time steps. As depicted in Fig. 9.3, the average reward achieved as a function of training steps shows that both policies learned the first phase of approaching the lesion quickly. However, Safe-PPO incurred a collision penalty at the start, which Unsafe-PPO did not. As a result, Safe-PPO remained lower in reward compared to Unsafe-PPO until 400 epochs. After 400 epochs, Safe-PPO correctly learned the trajectories to approach the lesion while avoiding unsafe configurations, leading to a higher reward. The rewards showed a significant increase at 800 epochs, which can be attributed to the learning of safe trajectories for the retract phase.

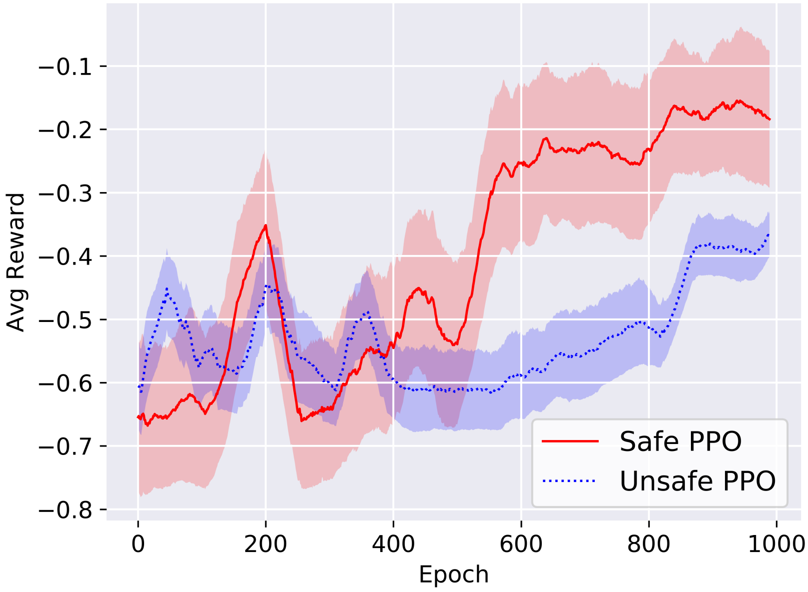


Fig. 9.3 The obtained learning curves for Safe-PPO and Unsafe PPO. The curves are averaged over four different seeds and smoothed over 25 epochs.

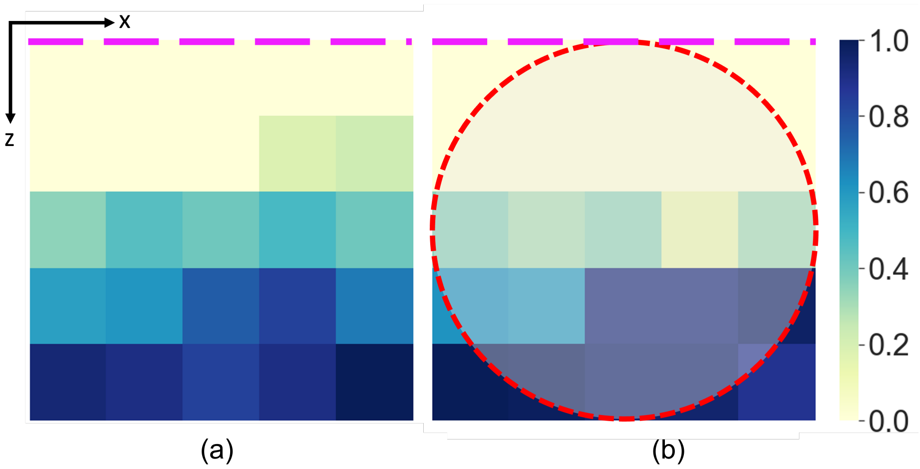


Fig. 9.4 Top-view of safe EE workspace showing the TE from different starting points (a) Unsafe-PPO (b) Safe-PPO. The marked circle shows the safe workspace projection, while the dashed pink line represents the attachment region. See text for more details.

Table 9.2 Summary of violation rates for each property

		Violation rate (%)					
Properties	**Property description**	**Safe-PPO**	**Unsafe-PPO**	**Primitive Safe-PPO**	**Policy4**	**Policy5**	**Policy6**
Θ_{1L}	Lower limit on x-direction (approach)	24.4	91.4	32.0	0.0	14.0	86.7
Θ_{1R}	Upper limit on x-direction (approach)	0.0	7.6	57.6	0.0	51.2	7.0
Θ_{2L}	Lower limit on y-direction (approach)	0.0	0.0	0.0	0.0	0.0	0.0
Θ_{3L}	Lower limit on z-direction (approach)	9.4	61.5	0.0	0.0	14.3	40.0
Θ_{3R}	Upper limit on z-direction (approach)	0.0	29.8	100.0	100.0	0.0	32.3
Θ_{4L}	Lower limit on x-direction (retract)	0.0	11.3	14.1	10.5	100.0	1.6
Θ_{4R}	Upper limit on x-direction (retract)	0.0	20.7	100.0	0.0	0.0	0.0
Θ_{5L}	Lower limit on y-direction (retract)	0.0	75.0	81.2	0.0	0.0	0.0
Θ_{5R}	Upper limit on y-direction (retract)	0.0	0.0	43.7	49.2	0.0	0.0
Θ_{6L}	Lower limit on z-direction (retract)	0.0	0.0	3.7	0.0	36.7	0.0
Θ_{6R}	Upper limit on z-direction (retract)	0.0	0.0	0.0	0.0	0.0	0.0
Average (Θ_{1L}-Θ_{2L})		8.14	33.01	29.88	0.0	21.74	31.25
Average (Θ_{3L}-Θ_{4R})		4.69	45.61	50.00	50.00	7.13	36.15
Average (Θ_{5L}-Θ_{6R})		0.00	17.84	40.46	9.96	22.79	0.26
Overall Average		**3.07**	27.02	39.31	14.52	19.65	15.24

The results of the violation rate for the ablation studies using FV are presented in Table 9.2. The policies that were trained are listed in Table 9.1. The mean global violation rate for Safe-PPO is 3.07%, while the mean violation rate for Unsafe-PPO is 27%. This demonstrates that incorporating safety criteria through the use of a collision penalty can significantly increase the safety of the procedure.

However, reporting the average violation rate alone does not provide a clear understanding of the distribution of violations, as some properties may be more critical than others in terms of the damage that can be caused if violated. For example, a significant proportion of the violations for Safe-PPO correspond to property Θ_{1L}. This highlights the difficulty in satisfying this property due to the presence of a complex obstacle, such as the spinal column, in that direction.

During the early stages of training, Primitive Safe-PPO incurs several collision penalties, which initially reduces the overall safety of the trajectory (refer to the third column in Table 9.2). As a result, it has a higher violation rate for several properties. Properties Θ_{2L} and Θ_{6R} have a 0% violation rate for all policies, demonstrating that all policies remain within the safety limits for these properties.

Policies 4, 5, and 6 consider a subset of properties during their training and have an average global violation rate that falls between that of Safe-PPO and Unsafe-PPO. These policies show 0% violations for the properties considered in their training, but have a significantly higher violation rate for other properties. This could be due to compensatory behavior, in which optimizing for one set of properties results in unsafe configurations for other properties. Further investigation is necessary to fully understand the high violation rates for certain properties.

In Safe-PPO, the states that can result in violation are represented by using a FV tool as shown in Fig. 9.5 (left), whereas the states encountered during standard execution are shown in Fig. 9.5 (right). The observation space consists of a 7-dimensional continuous input and a discrete input for grasping (as described in Sec. 9.2.1). In order to visualize the states in 2 dimensions, the values for the x and z motion of the EE are fixed by sampling from a normal distribution for each episode, and the FV tool is applied to the entire input domain of the EE movement in the y-direction and the target distance.

Fig. 9.5 (right) indicates a linear relationship between the EE movement in the y-direction and the target distance, while Fig. 9.5 (left) shows that the majority of state violations occur for lower values of EE Y in the range [0.0,0.2] and higher values of target distance in the range [0.5,0.8]. These violations are non-intuitive as they may be caused by violations in other state inputs, which are normally sampled. The figure demonstrates that Safe-PPO rarely encounters states that result in a violation. Even if such adversarial perturbations occur infrequently in a real-world robotic system, they can lead to fatal consequences. By using the proposed FV tool, these hazardous states can be identified in advance by the policy.

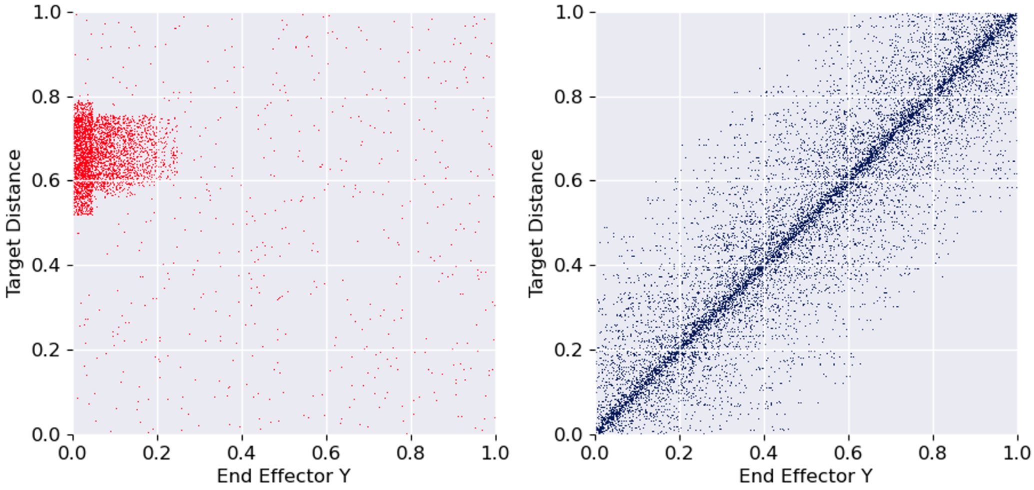


Fig. 9.5 (left) State values that cause a violation for Safe-PPO derived using the FV tool and (right) State distribution in a standard execution of Safe-PPO (1000 episodes). We describe the relationship between two-state inputs, i.e. normalized EE movement in the y-direction and target distance, to simplify visualization and use static values for other inputs.

It should be noted that incorporating safety constraints into the training loop does not increase computational time. The FV is an offline process carried out after training and does not impact the learned behavior. The TE matrix obtained for both Unsafe-PPO and Safe-PPO is shown in Fig. 9.4. The two methods show similar TE at all considered grid locations, with average TE being almost identical for both methods at 0.42 for Unsafe-PPO and 0.41 for Safe-PPO. This demonstrates that adding safety conditions does not affect overall task performance, providing a safety guarantee with optimal performance in terms of TE.

In the proximal region of the fat tissue attachment, the TE is low, corresponding to the upper area of the plots in Fig. 9.4. In this region, the EE grasps the fat tissue near the attachment and reaches the target position without any TE. This behavior is likely due to the fact that the reward function changes dramatically upon grasping the fat and does not penalize if the grasping point is far from the tumor. However, in the regions distal from the attachment, the grasping point comes closer to the tumor, thereby exposing the tumor. Future research will aim to improve this behavior by introducing the TE factor into the reward function.

9.4 Conclusions

In this study, we aim to mitigate the risks associated with actions taken during the training of DRL in safety-critical scenarios, such as surgical robotics. We propose the Safe-RL framework, which enables the incorporation of safety constraints through reward shaping. Additionally, we develop a FV tool to assess the extent of safety violations caused by a DRL policy. This tool allows us to identify states that may result in safety violations prior to model execution.

We demonstrate our approach by automating the task of TR, a common task in MIS, in a virtual environment. TR poses the risk of surrounding tissue damage if the robotic EE exceeds the workspace limits. To mitigate this risk, we design a safe workspace and add safety criteria for violations. Our results demonstrate increased safety and more reliable trajectories when using the safety protocol compared to traditional DRL methods without safety considerations.

In future work, we plan to implement the FV controller during model execution to prevent undesirable actions, and conduct experiments on a real robotic system using the simulation pipeline established in previous studies [52]. Further research is also needed to extend the applicability of the learned policies to different surgical scenarios. A key finding from this study is the importance of prioritizing different properties, as some may be more important than others. By assigning weights to different properties based on prior knowledge of the surgical scenario, it may be possible to achieve even safer behavior.

Contributions of this chapter

1. A Safe-RL framework for automating surgical subtasks, where safety problem can be encoded as a set of properties that provide limits to the safe workspace.
2. A FV tool for evaluating the violation of safety properties. This tool gives the probability of unsafe configurations over the designed set of properties.

Publications linked to this chapter

1. Ameya Pore, Davide Corsi, Enrico Marchesini, Diego Dall'Alba, Alicia Casals, Alessandro Farinelli, Paolo Fiorini. "Safe reinforcement learning using formal verification for TR in autonomous robotic-assisted surgery" In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3261-3266. IEEE, 2020.

Chapter 10

Conclusions and Future work

The primary objective of this thesis is to explore the potential of automating IP subtasks using data-driven approaches such as DRL to overcome the limitations of available clinical approaches.

IP involve the use of snake-like flexible instruments that present several maneuverability challenges. The continuous contact with the lumen wall and non-intuitive input-output relationship make the procedure challenging for human operators. In Chapter 2, the challenges with current IP instruments are discussed, highlighting the need for IP robotic assistance to reduce the physical and cognitive load on the clinicians.

The existing robotic systems in the market offer minimum autonomy, with the human operator controlling the robotic systems for the entire duration of the procedure. The thesis provides an overview of these commercially available IP robotic systems, highlighting their shortcomings. The learning curve for control systems is steep, requiring the clinicians to undergo extensive training processes with different sensorimotor feedback. Hence, the aim of this thesis is to increase the level of autonomy through subtask automation, placing the clinicians in a supervisory role.

One of the ways of incorporating automation is through low-level MP, which involves generating a smooth, collision-free, and optimal path for a robotic system to move from an initial configuration to a desired final configuration, while considering constraints and obstacles. An overview of the various MP methods for IP was presented in Chapter 4, including Node-based, Sampling-based, Optimization-based, and Learning-based methods. Our survey revealed that the recent advancements in DNN have led to an increased implementation of learning-based techniques, due to their ability to approximate non-linear functions. DRL has emerged as a suitable candidate for robotic automation. The hypothesis of this thesis is that DRL algorithms can reach near human performance on IP subtasks.

The validation of medical robotic systems is critical and safety is one of the most important considerations. In order to ensure that the robotic system behaves in a safe manner, it is a common practice to test the system in a simulated environment. There are several methods of simulation, including heuristic and continuum mechanical modeling [237], discussed in Chapter 4. However, it is important to find a balance between simulation accuracy and computation time. Hence, in this thesis, a real-time realistic characteristic simulation environment was developed for colonoscopy and robot-assisted MIS carried out using the dVSS. This simulation environment offers a realistic representation of the procedures, ensuring that the safety of the robotic system can be thoroughly evaluated and validated before it is used in a clinical setting.

Autonomous colonoscopy navigation has garnered significant attention as a promising approach for improving the accuracy and safety of endoscopic examinations. However, conventional techniques utilizing heuristic control policies have limitations in adapting to challenging scenarios where accurate detection of the colonic lumen becomes a major hindrance and requires frequent human intervention. In an effort to address these challenges, we present a novel approach for autonomous colonoscopy navigation using DVC. Our method leverages the ability of DVC to learn a mapping between endoscopic images and the control signal, thereby allowing for real-time control of the endoscope. To evaluate the effectiveness of our approach, we conducted a comprehensive performance comparison between our DVC control and motion data collected from 20 expert endoscopists.

The results of our evaluation showed that the performance of DVC control was equivalent to that of the expert endoscopists in terms of the time of insertion and the distance traveled. However, the DVC approach showed significant improvements in terms of reducing the number of colon wall collisions and efficient lumen tracking, thereby enhancing the safety of the examination. Additionally, a second novice user study was conducted to demonstrate the potential benefits of the DVC control in reducing user workload with overall performance comparable to that of expert endoscopists.

Our proposed DVC method open up avenues for further studies towards increasing the safety, accuracy and efficiency of the colonoscopy procedure with improved user workload compared to traditional heuristic control policies. We propose a constrained RL framework in which safety constraints could be added to avoid undesirable actions. Furthermore, we provide a model selection tool that can provide formal guarantees of a safe behavior.

Further, we present a novel approach towards an efficient and non-invasive solution for CRC tissue scans. Our proposed solution involves the integration of OCT with a robotic FE to provide an effective method for detecting and scanning abnormal tissue. To achieve this, we developed an autonomous robotic control strategy that leverages feedback from a monocular endoscopic camera and OCT imaging. The control strategy is formulated as an optimization problem, taking into account the orientation, depth and position of the endoscopic 2D image. This problem is then solved using a QP approach. Our approach demonstrates the feasibility of targeted OCT scanning and offers a potential solution for reducing the need for tissue biopsies. Additionally, we have validated our approach through experiments conducted in a synthetic colon environment, in varying lighting conditions.

The identification of malignant tissue requires a meticulous approach to ensure the safe removal of the tissue while minimizing any damage to the surrounding areas. The TR gesture is a critical aspect of the polypectomy procedure, which involves manipulating the tissue to retract and dissect it. To address the challenges associated with TR, we developed a simulation environment using the dVSS PSM to simulate the TR gesture. We showed that a DRL agent trained in simulation can be transferred to the real system with remarkable success. Additionally, we proposed a LfD methodology for TR automation based on GAIL, which enables the agent to learn from a small set of real demonstrations and be deployed in the real environment.

To address the safety concerns associated with DRL training, we introduced the Safe-DRL framework. This framework enables the addition of safety constraints through reward shaping and the evaluation of policy violations through FV. This allows us to identify potential states that may cause safety violations and prevent them from happening. The risks associated with TR mainly consist of surrounding tissue damage if the robotic end-effector exceeds the workspace limits. To mitigate this risk, we designed a safe workspace and added safety criteria to prevent workspace violations. Our results showed that incorporating safety protocols increased the safety of the TR task and improved the reliability of the trajectories performed using the Safe-DRL framework.

10.1 Future research directions

Reaching higher LoA in navigation requires accurate control, enhanced shape-sensing capabilities, tissue modeling capability and efficient MP. In this section, we discuss various missing capabilities in current IP robotic systems that hinder the development of a LoA 4 navigation system. Specifically, we divide the section in two parts. First we discuss the upcoming low-level technologies to enhance robotic capabilities. In the first part, we will examine the emerging low-level technologies that can enhance the capabilities of robots. Without advancements in these technologies, it would be challenging, if not impossible, to enhance MP. Second, we describe strategies to mitigate the challenges specific to applying DRL to real robot learning, since the major contribution of the thesis lies in implementing DRL based MP for IP subtask automation.

10.1.1 Robotic capabilities

In this subsection, we discuss the various actuation methods employed in continuum robots, including multi-link systems and soft robotics. We also explore the importance of proprioception and shape-sensing in achieving precise motion control, and the challenges associated with accurately modeling the lumen or vessels. Finally, we examine the role of intraoperative imaging modalities and Simultaneous Localization And Mapping (SLAM) in lumen/vessel modeling.

Robotics actuation

Continuum robots employed in IP procedures are developed based on different designs and technologies. For instance, several continuum instruments use concentric tube mechanisms or multi-link systems [38, 37]. Soft-robotics systems are an emerging paradigm that can enable multi-steering capabilities and complex stress-less interventions through narrow passageways [373]. IP scenarios reflect an environment where the snake-like robot can use the wall as a support to propel forward. Bio-inspired robots imitate biological systems such as snake locomotion [374, 375], octopus tentacles [376], elephant trunks [377], and mammalian spine [378]. Some studies have modeled the contact forces and friction for obstacle-aided dynamics of the snake [374]. Research on obstacle-aided locomotion can help to develop adaptive motion to operate in a constrained endoluminal environment. Some early robotic prototypes include flexible joint mechanism prototype [378], and a tendon-driven snake robot [379]. Furthermore, some works have proposed MP algorithms for serpentine robots [380, 375]. Elephant trunk models have been of interest in the field of soft hyper-redundant robots. The standard structural design includes a trunk backbone with multiple segments [381, 377]. Another model for bio-inspiration is the Octopus tentacles. A recent study has developed prototypes using fluid actuators that mimic the octopus tentacle behavior [376]. Pressure-driven eversion of flexible, thin-walled tubes, called vine robots, has shown increased applications to navigate confined spaces [382].

Proprioception and Shape-sensing

To achieve precise and reliable motion control of continuum robots, accurate and real-time shape sensing is needed. However, accurately modeling the robot shape is challenging due to friction, backlash and the inherent deformable nature of the lumen or vessels and inevitable collisions with the anatomy [383]. Some emerging techniques for shape reconstruction together with tip localization rely on Fiber Bragg Gratings (FBG) and EM sensors [383–385]. FBG-enabled sensing techniques can provide real-time force measurement and shape estimation without requiring kinematic-based modeling. Multiple miniature EM sensors attached along the continuum robot have been applied to track and localize the robot. Moreover, computer vision techniques can be utilized to estimate the pose of the robot [386].

Lumen/vessel modeling

Intraoperative imaging modalities such as ultrasound and optical computed tomography can support direct observation and visualization [387, 316]. For computer-assisted navigation, SLAM has been successfully demonstrated in inferring dense and detailed depth maps and lumen reconstruction [76]. Depth prediction models are developed recently to estimate lumen features [388].

10.1.2 Outstanding challenges in DRL and strategies for its mitigation

Simulation

While we show that learning based methods such as DRL can be used for surgical subtask automation, one of the major bottleneck in the successful deployment of DRL is the need for a highly accurate simulated environment that perfectly resembles an open-world environment. While collecting enough real data on the physical system is slow and expensive, simulation can run orders of magnitude faster than real-time, and can start many instances simultaneously. However, it is challenging to generalize the knowledge gained through training in a simulator to a real situation, called the “sim-to-real" reality gap due to the discrepancies between reality and virtual environment that occur due to modeling errors [389]. This issue becomes particularly significant when working with image-based approaches due to the large visual domain gap between simulated and real images. Various methodologies for bridging this gap and transferring image-based policies are explored. One such approach is Domain Randomization (DR), which mitigates the sim-to-real gap by introducing random variations to visual parameters in the simulation, such as texture and lighting. This allows the trained policy to learn generalized and task-relevant visual features. While DR approaches have demonstrated successful translation into reality, they can be challenging to tune and may be highly specific to certain tasks [390]. Recent studies have attempted to bridge this visual sim-to-real gap with an image-based DRL pipeline based on pixel-level domain adaptation using methods such as Cycle-GAN and contrastive unsupervised translation [349].

Currently, in the development of an image-based navigation policy based on DVC (discussed in Chapter 5), the validation was carried out in a virtually simulated domain. The goal of our future works would be to deploy the method in a real robotized FE such as STRAS with realistic clinical scenarios. This would require implementing the above mentioned sim-to-real techniques.

Sample efficiency

Some classes of RL algorithms are much more efficient that others. RL algorithms can be categorized into model-based versus model-free methods. Model-based algorithms choose optimal action by leveraging a model of the environment. The agent may learn from the experience generated using this model instead of collected in the real environment. Thus the amount of data required for model-based methods is usually much less than their model-free counterparts. The downside is that these methods require to have access to such a model, which is often challenging to acquire in practice. For example: tissue deformations are challenging to model. One future direction would be to train a deformation model of the tissue using neural network such as U-Mesh [391]. This model can be further used to train DRL agents.

In model-based RL, demonstration data can also be aggregated with the agent's experience to produce better models. However, in contrast to the model-free setting, for model-based RL this approach can be quite effective, because it would enable the learned model to capture correct dynamics in important parts of the state space. When combined with a good planning method, which can also use the demonstrations (e.g., as a proposal distribution), including demonstrations into the model training dataset can enable a robot to perform complex behaviors which would be extremely difficult to discover automatically [392].

The DRL methods used in this thesis are on-policy algorithms that use a sample coming from the latest policy that is being trained. Offline training offers potential as a large volume of data can be used to pre-train the robot. In such a setting, samples can be reused multiple times across back-propagations, hundreds or thousands of times without any over-fitting in complex visual tasks.

With this thesis, we have shown that subtask automation of IP can greatly reduce the workload of the clinicians. Learning-based methods are one of the potential candidates to develop adaptable and task generalizable surgical skills. Despite the promising results already obtained, the methods proposed in this thesis offer margin for improvement such as

1. Improved task generalization: One of the key challenges in surgical robotics is the need to generalize learned policies to new surgical scenarios. There is a need to develop algorithms that can handle novel scenarios, including novel anatomies, instruments, and surgical tasks. This requires designing more complex reward functions that capture the nuances of surgical performance.

2. Real-time safety guarantees: Safety is a critical concern in surgical robotics. It is essential to develop algorithms that can provide real-time safety guarantees and prevent dangerous actions. This requires designing algorithms that can learn safe policies while minimizing negative interactions with the environment.

3. Multi-modal sensing and perception: Robotic systems should be able to sense and perceive the surgical environment in real-time. There is a need to develop algorithms that can handle different modalities of sensing, including vision, haptics, and other forms of sensory information.

4. Adaptive control strategies: The dynamics of surgical procedures are highly variable and can change quickly. There is a need to develop adaptive control strategies that can handle these variations and maintain stable control over the robot.

5. Human-robot collaboration: Finally, there is a need to develop algorithms that can enable effective human-robot collaboration in surgical procedures. This requires design-

ing algorithms that can handle natural language commands, understand the surgeon's intent, and adapt to changes in the surgical plan.

These improvement could bring the applicability of autonomous systems closer to real clinical conditions, thus opening space for new exciting research directions.

References

[1] A. Attanasio, B. Scaglioni, E. De Momi, P. Fiorini, and P. Valdastri, "Autonomy in surgical robotics," Annual Review of Control, Robotics, and Autonomous Systems, vol. 4, 2020.

[2] P. T. Tran, G. Smoljkic, C. Gruijthuijsen, D. Reynaerts, J. Vander Sloten, and E. Vander Poorten, "Position control of robotic catheters inside the vasculature based on a predictive minimum energy model," in 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2016, pp. 004 687–004 693.

[3] H. Sharei, T. Alderliesten, J. J. van den Dobbelsteen, and J. Dankelman, "Navigation of guidewires and catheters in the body during intervention procedures: a review of computer-based models," Journal of Medical Imaging, vol. 5, no. 1, p. 010902, 2018.

[4] W. Tang, P. Lagadec, D. Gould, T. R. Wan, J. Zhai, and T. How, "A realistic elastic rod model for real-time simulation of minimally invasive vascular interventions," The Visual Computer, vol. 26, no. 9, pp. 1157–1165, 2010.

[5] M. Peral-Boiza, T. Gomez-Fernandez, P. Sanchez-Gonzalez, B. Rodriguez-Vila, E. J. Gómez, and Á. Gutiérrez, "Position based model of a flexible ureterorenoscope in a virtual reality training platform for a minimally invasive surgical robot," IEEE Access, vol. 7, pp. 177 414–177 426, 2019.

[6] V. Luboz, R. Blazewski, D. Gould, and F. Bello, "Real-time guidewire simulation in complex vascular models," The Visual Computer, vol. 25, no. 9, pp. 827–834, 2009.

[7] Y. Wei, S. Cotin, J. Dequidt, C. Duriez, J. Allard, E. Kerrien et al., "A (near) real-time simulation method of aneurysm coil embolization," Aneurysm, vol. 8, no. 29, pp. 223–248, 2012.

[8] H. Jung, D. Y. Lee, and W. Ahn, "Real-time deformation of colon and endoscope for colonoscopy simulation," The International Journal of Medical Robotics and Computer Assisted Surgery, vol. 8, no. 3, pp. 273–281, 2012.

[9] H. De Visser, J. Passenger, D. Conlan, C. Russ, D. Hellier, M. Cheng, O. Acosta, S. Ourselin, and O. Salvado, "Developing a next generation colonoscopy simulator," International Journal of Image and Graphics, vol. 10, no. 02, pp. 203–217, 2010.

[10] Z. Qiukui and P. Haigron, "A fem model for interactive simulation of guide wire navigation in moving vascular structures," in 2015 Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA). IEEE, 2015, pp. 13–16.

[11] J.-P. Zheng, C.-Z. Li, G.-Q. Chen, G.-D. Song, and Y.-Z. Zhang, "Three-dimensional printed skull base simulation for transnasal endoscopic surgical training," World neurosurgery, vol. 111, pp. e773–e782, 2018.

[12] S. Athiniotis, R. Srivatsan, and H. Choset, "Deep q reinforcement learning for autonomous navigation of surgical snake robot in confined spaces," in The Hamlyn Symposium on Medical Robotics, 2019.

[13] W. Chi et al., "Trajectory optimization of robot-assisted endovascular catheterization with reinforcement learning," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, Oct. 2018.

[14] J.-Q. Zheng, X.-Y. Zhou, C. Riga, and G.-Z. Yang, "Towards 3d path planning from a single 2d fluoroscopic image for robot assisted fenestrated endovascular aortic repair," in International Conference on Robotics and Automation (ICRA). IEEE, pp. 8747–8753, 2019.

[15] A. Follmann, C. B. Pereira, J. Knauel, R. Rossaint, and M. Czaplik, "Evaluation of a bronchoscopy guidance system for bronchoscopy training, a randomized controlled trial," BMC medical education, vol. 19, no. 1, pp. 1–7, 2019.

[16] C. J. Laborde, C. S. Bell, J. C. Slaughter, P. Valdastri, and K. L. Obstein, "Evaluation of a novel tablet application for improvement in colonoscopy training and mentoring (with video)," Gastrointestinal endoscopy, vol. 85, no. 3, pp. 559–565, 2017.

[17] K. İncetan, I. O. Celik, A. Obeid, G. I. Gokceler, K. B. Ozyoruk, Y. Almalioglu, R. J. Chen, F. Mahmood, H. Gilbert, N. J. Durr et al., "Vr-caps: A virtual environment for capsule endoscopy," Medical image analysis, vol. 70, p. 101990, 2021.

[18] V. Vitiello, S.-L. Lee, T. P. Cundy, and G.-Z. Yang, "Emerging robotic platforms for minimally invasive surgery," IEEE reviews in biomedical engineering, vol. 6, pp. 111–126, 2012.

[19] M. A. REUTER and H. J. REUTER, "The development of the cystoscope," The Journal of urology, vol. 159, no. 3, pp. 638–640, 1998.

[20] G. S. Litynski, "Laparoscopy-the early attempts: Spotlighting georg kelling and hans christian jacobaeus," JSLS: Journal of the Society of Laparoendoscopic Surgeons, vol. 1, no. 1, p. 83, 1997.

[21] N. Simaan, R. M. Yasin, and L. Wang, "Medical technologies and challenges of robot-assisted minimally invasive intervention and diagnostics," Annual Review of Control, Robotics, and Autonomous Systems, vol. 1, pp. 465–490, 2018.

[22] A. Orekhov, C. Abah, and N. Simaan, "Snake-like robots for minimally invasive, single-port, and intraluminal surgeries," The Enciclopedia of Medical Robotics. World Scientific, pp. 203–243, 2018.

[23] J. Seetohul et al., "Snake robots for surgical applications: A review," Robot., vol. 11, no. 3, p. 57, 2022.

[24] T. da Veiga, J. H. Chandler, P. Lloyd, G. Pittiglio, N. J. Wilkinson, A. K. Hoshiar, R. A. Harris, and P. Valdastri, "Challenges of continuum robots in clinical context: a review." Progress in Biomedical Engineering, 2020.

[25] G.-Z. Yang, J. Cambias, K. Cleary, E. Daimler, J. Drake, P. E. Dupont, N. Hata, P. Kazanzides, S. Martel, R. V. Patel et al., "Medical robotics—regulatory, ethical, and legal considerations for increasing levels of autonomy," Science Robotics, vol. 2, no. 4, p. 8638, 2017.

[26] J. M. Prendergast, G. A. Formosa, C. R. Heckman, and M. E. Rentschler, "Autonomous localization, navigation and haustral fold detection for robotic endoscopy," in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018, pp. 783–790.

[27] J. Hwang, J.-y. Kim, and H. Choi, "A review of magnetic actuation systems and magnetically actuated guidewire-and catheter-based microrobots for vascular interventions," Intelligent Service Robotics, vol. 13, no. 1, pp. 1–14, 2020.

[28] L. Manfredi, "Endorobots for colonoscopy: Design challenges and available technologies," Frontiers in Robotics and AI, p. 209, 2021.

[29] R. Hargest, "Five thousand years of minimal access surgery: 1990–present: organisational issues and the rise of the robots," Journal of the Royal Society of Medicine, vol. 114, no. 2, pp. 69–76, 2021.

[30] T. Hu, P. K. Allen, N. J. Hogle, and D. L. Fowler, "Insertable surgical imaging device with pan, tilt, zoom, and lighting," The International Journal of Robotics Research, vol. 28, no. 10, pp. 1373–1386, 2009.

[31] E. A. Arkenbout, P. W. Henselmans, F. Jelínek, and P. Breedveld, "A state of the art review and categorization of multi-branched instruments for notes and sils," Surgical endoscopy, vol. 29, pp. 1281–1296, 2015.

[32] P. Fiorini, K. Y. Goldberg, Y. Liu, and R. H. Taylor, "Concepts and trends in autonomy for robot-assisted surgery," Proceedings of the IEEE, vol. 110, no. 7, pp. 993–1011, 2022.

[33] T. Haidegger, "Autonomy for surgical robots: Concepts and paradigms," IEEE Transactions on Medical Robotics and Bionics, vol. 1, no. 2, pp. 65–76, 2019.

[34] M.-C. Fiazza and P. Fiorini, "Design for interpretability: Meeting the certification challenge for surgical robots," in IEEE International Conference on Intelligence and Safety for Robotics (ISR). IEEE, 2021, pp. 264–267.

[35] B. Patle, A. Pandey, D. Parhi, A. Jagadeesh et al., "A review: On path planning strategies for navigation of mobile robot," Defence Technology, vol. 15, no. 4, pp. 582–606, 2019.

[36] J.-C. Latombe, Robot motion planning. Springer Science & Business Media, vol. 124, 2012.

[37] O. M. Omisore, S. Han, J. Xiong, H. Li, Z. Li, and L. Wang, "A review on flexible robotic systems for minimally invasive surgery," IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2020.

[38] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," IEEE Transactions on Robotics, vol. 31, no. 6, pp. 1261–1280, 2015.

[39] R. S. Sutton and A. G. Barto, Reinforcement learning: An introduction. MIT press, 2018.

[40] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," arXiv preprint arXiv:1312.5602, 2013.

[41] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot et al., “Mastering the game of go with deep neural networks and tree search,” nature, vol. 529, no. 7587, p. 484, 2016.

[42] T. Haarnoja, S. Ha, A. Zhou, J. Tan, G. Tucker, and S. Levine, “Learning to walk via deep reinforcement learning,” arXiv preprint arXiv:1812.11103, 2018.

[43] S. Gu, E. Holly, T. Lillicrap, and S. Levine, “Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates,” in 2017 IEEE international conference on robotics and automation (ICRA). IEEE, 2017, pp. 3389–3396.

[44] T. Haarnoja, V. Pong, A. Zhou, M. Dalal, P. Abbeel, and S. Levine, “Composable deep reinforcement learning for robotic manipulation,” in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 6244–6251.

[45] J. Ibarz, J. Tan, C. Finn, M. Kalakrishnan, P. Pastor, and S. Levine, “How to train your robot with deep reinforcement learning: lessons we have learned,” The International Journal of Robotics Research, vol. 40, no. 4-5, pp. 698–721, 2021.

[46] S. Datta, Y. Li, M. M. Ruppert, Y. Ren, B. Shickel, T. Ozrazgat-Baslanti, P. Rashidi, and A. Bihorac, “Reinforcement learning in surgery,” Surgery, vol. 170, no. 1, pp. 329–332, 2021.

[47] Z.-Y. Chiu, F. Richter, E. K. Funk, R. K. Orosco, and M. C. Yip, “Bimanual regrasping for suture needles using reinforcement learning for rapid motion planning,” in 2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2021, pp. 7737–7743.

[48] T. Osa, N. Sugita, and M. Mitsuishi, “Online trajectory planning and force control for automation of surgical tasks,” IEEE Transactions on Automation Science and Engineering, vol. 15, no. 2, pp. 675–691, 2017.

[49] A. Murali, S. Sen, B. Kehoe, A. Garg, S. McFarland, S. Patil, W. D. Boyd, S. Lim, P. Abbeel, and K. Goldberg, “Learning by observation for surgical subtasks: Multilateral cutting of 3d viscoelastic and 2d orthotropic tissue phantoms,” in 2015 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2015, pp. 1202–1209.

[50] C. Shin, P. W. Ferguson, S. A. Pedram, J. Ma, E. P. Dutson, and J. Rosen, “Autonomous tissue manipulation via surgical robot using learning based model predictive control,” in 2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019, pp. 3875–3881.

[51] F. Richter, R. K. Orosco, and M. C. Yip, “Open-sourced reinforcement learning environments for surgical robotics,” arXiv preprint arXiv:1903.02090, 2019.

[52] E. Tagliabue, A. Pore, D. Dall’Alba, E. Magnabosco, M. Piccinelli, and P. Fiorini, “Soft tissue simulation environment to learn manipulation tasks in autonomous robotic surgery,” in 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2020, pp. 3261–3266.

[53] A. Pore, E. Tagliabue, M. Piccinelli, D. Dall’Alba, A. Casals, and P. Fiorini, “Learning from demonstrations for autonomous soft-tissue retraction,” in 2021 International Symposium on Medical Robotics (ISMR). IEEE, 2021, pp. 1–7.

[54] N. v. d. Stap, C. H. Slump, I. A. Broeders, and F. v. d. Heijden, "Image-based navigation for a robotized flexible endoscope," in International Workshop on Computer-Assisted and Robotic Endoscopy. Springer, 2014, pp. 77–87.

[55] J. Lee, "Resection of diminutive and small colorectal polyps: What is the optimal technique?" Clinical endoscopy, vol. 49, no. 4, pp. 355–358, 2016.

[56] S. Nivatvongs, "Surgical management of malignant colorectal polyps," Surgical Clinics, vol. 82, no. 5, pp. 959–966, 2002.

[57] Y. Zeng, S. Xu, W. C. Chapman, S. Li, Z. Alipour, H. Abdelal, D. Chatterjee, M. Mutch, and Q. Zhu, "Real-time colorectal cancer diagnosis using pr-oct with deep learning," in Optical Coherence Tomography. Optical Society of America, 2020, pp. OW2E–5.

[58] A. Attanasio, B. Scaglioni, M. Leonetti, A. F. Frangi, W. Cross, C. S. Biyani, and P. Valdastri, "Autonomous tissue retraction in robotic assisted minimally invasive surgery–a feasibility study," IEEE Robotics and Automation Letters, vol. 5, no. 4, pp. 6528–6535, 2020.

[59] S. Y. Nof, "Automation: What it means to us around the world," in Springer handbook of automation. Springer, 2009, pp. 13–52.

[60] J. I. Olszewska, M. Barreto, J. Bermejo-Alonso, J. Carbonera, A. Chibani, S. Fiorini, P. Goncalves, M. Habib, A. Khamis, A. Olivares et al., "Ontology for autonomous robotics," in 26th IEEE international symposium on robot and human interactive communication (RO-MAN). IEEE, pp. 189–194, 2017.

[61] H. Chen, Y. Wen, M. Zhu, Y. Huang, C. Xiao, T. Wei, and A. Hahn, "From automation system to autonomous system: An architecture perspective," Journal of Marine Science and Engineering, vol. 9, no. 6, p. 645, 2021.

[62] M. Fisher, V. Mascardi, K. Y. Rozier, B.-H. Schlingloff, M. Winikoff, and N. Yorke-Smith, "Towards a framework for certification of reliable autonomous systems," Autonomous Agents and Multi-Agent Systems, vol. 35, no. 1, pp. 1–65, 2021.

[63] F. Merenda, D. Nardi, F. Silvestri, A. Maria Zanchettin, E. Girardi, G. Bottos, A. Cianciosi, and A. Puligheddu, "Ethics, safety and human centricity: Intelligent machines under the scope of the european ai regulation act," in Workshop 7, 3rd Conference of the Italian Institute of Robotics and Intelligent Machines. IEEE, 2021.

[64] S. O'sullivan et al., "Legal, regulatory, and ethical frameworks for development of standards in artificial intelligence (ai) and autonomous robotic surgery," The International Journal of Medical Robotics and Computer Assisted Surgery, vol. 15, no. 1, p. e1968, 2019.

[65] "National artificial intelligence initiative," https://www.congress.gov/bill/116th-congress/house-bill/6216/text.

[66] E. Parliament and C. of the European Union, "Artificial intelligence act: Regulation laying down harmonised rules on artificial intelligence and amending certain union legislative acts," Proposal for Regulation COM/2021/206 final, Brussels, Belgium, 2021.

[67] M.-C. Fiazza, "The eu proposal for regulating ai: Foreseeable impact on medical robotics," in IEEE International Conference on advanced robotics (ICAR). IEEE, 2021.

[68] I. E. C. (2017c)., "Iec tr 60601-4-1 – medical electrical equipment – part 4-1: Guidance and interpretation - medical electrical equipment and medical electrical systems employing a degree of autonomy," in URL https://webstore.iec.ch/publication/29312.), 2017.

[69] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang, "Xai—explainable artificial intelligence," Science robotics, vol. 4, no. 37, p. eaay7120, 2019.

[70] D. Katić, C. Julliard, A.-L. Wekerle, H. Kenngott, B. P. Müller-Stich, R. Dillmann, S. Speidel, P. Jannin, and B. Gibaud, "Lapontospm: an ontology for laparoscopic surgeries and its application to surgical phase recognition," International journal of computer assisted radiology and surgery, vol. 10, no. 9, pp. 1427–1434, 2015.

[71] R. Khare, R. Bascom, and W. E. Higgins, "Hands-free system for bronchoscopy planning and guidance," IEEE Transactions on Biomedical Engineering, vol. 62, no. 12, pp. 2794–2811, 2015.

[72] X. Zang, J. D. Gibbs, R. Cheirsilp, P. D. Byrnes, J. Toth, R. Bascom, and W. E. Higgins, "Optimal route planning for image-guided ebus bronchoscopy," Computers in biology and medicine, vol. 112, p. 103361, 2019.

[73] A. Z. Taddese, P. R. Slawinski, M. Pirotta, E. De Momi, K. L. Obstein, and P. Valdastri, "Enhanced real-time pose estimation for closed-loop robotic manipulation of magnetically actuated capsule endoscopes," The International journal of robotics research, vol. 37, no. 8, pp. 890–911, 2018.

[74] K. A. Tran, O. Kondrashova, A. Bradley, E. D. Williams, J. V. Pearson, and N. Waddell, "Deep learning in cancer diagnosis, prognosis and treatment selection," Genome Medicine, vol. 13, no. 1, pp. 1–17, 2021.

[75] L. R. Kennedy-Metz, P. Mascagni, A. Torralba, R. D. Dias, P. Perona, J. A. Shah, N. Padoy, and M. A. Zenati, "Computer vision in the operating room: Opportunities and caveats," IEEE transactions on medical robotics and bionics, vol. 3, no. 1, pp. 2–10, 2020.

[76] F. Chadebecq, F. Vasconcelos, E. Mazomenos, and D. Stoyanov, "Computer vision in the surgical operating room," Visceral Medicine, vol. 36, no. 6, pp. 456–462, 2020.

[77] B. S. Peters, P. R. Armijo, C. Krause, S. A. Choudhury, and D. Oleynikov, "Review of emerging surgical robotic technology," Surgical endoscopy, vol. 32, no. 4, pp. 1636–1655, 2018.

[78] H. Rafii-Tari, C. J. Payne, and G.-Z. Yang, "Current and emerging robot-assisted endovascular catheterization technologies: a review," Annals of biomedical engineering, vol. 42, no. 4, pp. 697–715, 2014.

[79] J. Bonatti, G. Vetrovec, C. Riga, O. Wazni, and P. Stadler, "Robotic technology in cardiovascular medicine," Nature Reviews Cardiology, vol. 11, no. 5, p. 266, 2014.

[80] Y. Fu, H. Liu, W. Huang, S. Wang, and Z. Liang, "Steerable catheters in minimally invasive vascular surgery," The International Journal of Medical Robotics and Computer Assisted Surgery, vol. 5, no. 4, pp. 381–391, 2009.

[81] A. Pourdjabbar et al., "The development of robotic technology in cardiac and vascular interventions," Rambam Maimonides Med. J., vol. 8, no. 3, 2017.

[82] M. Berczeli et al., “Catheter robots in the cardiovascular system,” Latest Develop. Med. Robot. Syst., p. 95, 2021.

[83] G. Ciuti, K. Skonieczna-Żydecka, W. Marlicz, V. Iacovacci, H. Liu, D. Stoyanov, A. Arezzo, M. Chiurazzi, E. Toth, H. Thorlacius et al., “Frontiers of robotic colonoscopy: A comprehensive review of robotic colonoscopes and technologies,” Journal of Clinical Medicine, vol. 9, no. 6, p. 1648, 2020.

[84] C.-K. Yeung, J. L. Cheung, and B. Sreedhar, “Emerging next-generation robotic colonoscopy systems towards painless colonoscopy,” Journal of digestive diseases, vol. 20, no. 4, pp. 196–205, 2019.

[85] H. You, E. Bae, Y. Moon, J. Kweon, and J. Choi, “Automatic control of cardiac ablation catheter with deep reinforcement learning method,” Journal of Mechanical Science and Technology, vol. 33, no. 11, pp. 5415–5423, 2019.

[86] M. Lemke et al., “Colonoscopy trainers experience greater stress during insertion than withdrawal: implications for endoscopic curricula,” J. Can. Assoc. Gastroenterology, vol. 4, no. 1, pp. 15–20, 2021.

[87] R. Ahmed, K. Santhirakumar, H. Butt, and A. K. Yetisen, “Colonoscopy technologies for diagnostics and drug delivery,” Medical Devices & Sensors, vol. 2, no. 3-4, p. e10041, 2019.

[88] K. J. Wernli et al., “Risks associated with anesthesia services during colonoscopy,” Gastroenterology, vol. 150, no. 4, pp. 888–894, 2016.

[89] C. Hassan et al., “Diagnostic yield and miss rate of endorings in an organized colorectal cancer screening program: the smart (study methodology for adr-related technology) trial,” Gastrointestinal Endoscopy, vol. 89, no. 3, pp. 583–590, 2019.

[90] A. Eickhoff, R. Jakobs, A. Kamal, S. Mermash, J. Riemann, and J. Van Dam, “In vitro evaluation of forces exerted by a new computer-assisted colonoscope (the neoguide endoscopy system),” Endoscopy, vol. 38, no. 12, pp. 1224–1229, 2006.

[91] N. Gluck, A. Melhem, Z. Halpern, K. Mergener, and E. Santo, “A novel self-propelled disposable colonoscope is effective for colonoscopy in humans (with video),” Gastrointestinal endoscopy, vol. 83, no. 5, pp. 998–1004, 2016.

[92] M. Shike, Z. Fireman, R. Eliakim, O. Segol, A. Sloyer, L. B. Cohen, S. Goldfarb-Albak, and A. Repici, “Sightline colonosight system for a disposable, power-assisted, non-fiber-optic colonoscopy (with video),” Gastrointestinal endoscopy, vol. 68, no. 4, pp. 701–710, 2008.

[93] E. Tumino, G. Parisi, M. Bertoni, M. Bertini, S. Metrangolo, E. Ierardi, R. Cervelli, G. Bresci, and R. Sacco, “Use of robotic colonoscopy in patients with previous incomplete colonoscopy,” Eur Rev Med Pharmacol Sci, vol. 21, no. 4, pp. 819–826, 2017.

[94] S. D. Herrell, R. Webster, and N. Simaan, “Future robotic platforms in urologic surgery: recent developments,” Current opinion in urology, vol. 24, no. 1, p. 118, 2014.

[95] M. D. Tyson et al., “Urological applications of natural orifice transluminal endoscopic surgery,” Nature Rev. Urology, vol. 11, no. 6, pp. 324–332, 2014.

[96] W. M. Bazzi et al., “Natural orifice transluminal endoscopic surgery in urology: Review of the world literature,” Urology Ann., vol. 4, no. 1, p. 1, 2012.

[97] Y. Chen, S. Zhang, Z. Wu, B. Yang, Q. Luo, and K. Xu, “Review of surgical robotic systems for keyhole and endoscopic procedures: state of the art and perspectives,” Frontiers of medicine, vol. 14, no. 4, pp. 382–403, 2020.

[98] J. Rassweiler, M. Fiedler, N. Charalampogiannis, A. S. Kabakci, R. Saglam, and J.-T. Klein, “Robot-assisted flexible ureteroscopy: an update,” Urolithiasis, vol. 46, no. 1, pp. 69–77, 2018.

[99] G. Gandaglia et al., “Novel technologies in urologic surgery: a rapidly changing scenario,” Current urology Rep., vol. 17, no. 3, pp. 1–8, 2016.

[100] P. Valdastri, M. Simi, and R. J. Webster III, “Advanced technologies for gastrointestinal endoscopy,” Annual review of biomedical engineering, vol. 14, 2012.

[101] W. Marlicz, X. Ren, A. Robertson, K. Skonieczna-Żydecka, I. Łoniewski, P. Dario, S. Wang, J. N. Plevris, A. Koulaouzidis, and G. Ciuti, “Frontiers of robotic gastroscopy: A comprehensive review of robotic gastroscopes and technologies,” Cancers, vol. 12, no. 10, p. 2775, 2020.

[102] A. De Virgilio, A. Costantino, G. Mercante, P. Di Maio, O. Iocca, and G. Spriano, “Trans-oral robotic surgery in the management of parapharyngeal space tumors: a systematic review,” Oral Oncology, vol. 103, p. 104581, 2020.

[103] U. B. Prakash and R. A. Matthay, “Bronchoscopy,” Journal of Bronchology & Interventional Pulmonology, vol. 1, no. 4, p. 340, 1994.

[104] A. Agrawal et al., “Robotic bronchoscopy for pulmonary lesions: a review of existing technologies and clinical data,” J. thoracic disease, vol. 12, no. 6, p. 3279, 2020.

[105] A. B. Villaret, F. Doglietto, A. Carobbio, A. Schreiber, C. Panni, E. Piantoni, G. Guida, M. M. Fontanella, P. Nicolai, and R. Cassinis, “Robotic transnasal endoscopic skull base surgery: systematic review of the literature and report of a novel prototype for a hybrid system (brescia endoscope assistant robotic holder),” World neurosurgery, vol. 105, pp. 875–883, 2017.

[106] H. Stammberger and W. Posawetz, “Functional endoscopic sinus surgery,” European archives of oto-rhino-laryngology, vol. 247, no. 2, pp. 63–76, 1990.

[107] J. Burgner, D. C. Rucker, H. B. Gilbert, P. J. Swaney, P. T. Russell, K. D. Weaver, and R. J. Webster, “A telerobotic system for transnasal surgery,” IEEE/ASME Transactions on Mechatronics, vol. 19, no. 3, pp. 996–1006, 2013.

[108] A. Madoglio, F. Zappa, D. Mattavelli, V. Rampinelli, M. Ferrari, A. Schreiber, F. Belotti, A. B. Villaret, F. Tampalini, R. Cassinis et al., “Robotics in endoscopic transnasal skull base surgery: literature review and personal experience,” Control Systems Design of Bio-Robotics and Bio-mechatronics with Advanced Applications, pp. 221–244, 2020.

[109] N. Simaan, K. Xu, W. Wei, A. Kapoor, P. Kazanzides, R. Taylor, and P. Flint, “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.

[110] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” nature, vol. 521, no. 7553, pp. 436–444, 2015.

[111] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, “Deep reinforcement learning: A brief survey,” IEEE Signal Processing Magazine, vol. 34, no. 6, pp. 26–38, 2017.

[112] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski et al., “Human-level control through deep reinforcement learning,” nature, vol. 518, no. 7540, pp. 529–533, 2015.

[113] M. Campbell, A. J. Hoane Jr, and F.-h. Hsu, “Deep blue,” Artificial intelligence, vol. 134, no. 1-2, pp. 57–83, 2002.

[114] B. Thananjeyan, A. Garg, S. Krishnan, C. Chen, L. Miller, and K. Goldberg, “Multilateral surgical pattern cutting in 2d orthotropic gauze with deep reinforcement learning policies for tensioning,” in 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017, pp. 2371–2378.

[115] X. Wang, S. Wang, X. Liang, D. Zhao, J. Huang, X. Xu, B. Dai, and Q. Miao, “Deep reinforcement learning: a survey,” IEEE Transactions on Neural Networks and Learning Systems, 2022.

[116] R. Bellman, “On the theory of dynamic programming,” Proceedings of the national Academy of Sciences, vol. 38, no. 8, pp. 716–719, 1952.

[117] C. J. Watkins and P. Dayan, “Q-learning,” Machine learning, vol. 8, no. 3, pp. 279–292, 1992.

[118] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, “Asynchronous methods for deep reinforcement learning,” in International conference on machine learning. PMLR, 2016, pp. 1928–1937.

[119] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” arXiv preprint arXiv:1506.02438, 2015.

[120] M. P. Deisenroth, G. Neumann, J. Peters et al., “A survey on policy search for robotics,” Foundations and Trends® in Robotics, vol. 2, no. 1–2, pp. 1–142, 2013.

[121] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, “Evolution strategies as a scalable alternative to reinforcement learning,” arXiv preprint arXiv:1703.03864, 2017.

[122] E. Galván and P. Mooney, “Neuroevolution in deep neural networks: Current trends and future challenges,” IEEE Transactions on Artificial Intelligence, vol. 2, no. 6, pp. 476–493, 2021.

[123] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, “Trust region policy optimization,” in International conference on machine learning, 2015, pp. 1889–1897.

[124] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” arXiv preprint arXiv:1509.02971, 2015.

[125] N. Heess, G. Wayne, D. Silver, T. Lillicrap, T. Erez, and Y. Tassa, “Learning continuous control policies by stochastic value gradients,” Advances in neural information processing systems, vol. 28, 2015.

[126] J. Koutník, G. Cuccu, J. Schmidhuber, and F. Gomez, "Evolving large-scale neural networks for vision-based reinforcement learning," in Proceedings of the 15th annual conference on Genetic and evolutionary computation, 2013, pp. 1061–1068.

[127] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," Advances in neural information processing systems, vol. 12, 1999.

[128] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," Machine learning, vol. 8, no. 3, pp. 229–256, 1992.

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

[130] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in International conference on machine learning. PMLR, 2014, pp. 387–395.

[131] R. Bellman, "Dynamic programming," science, vol. 153, no. 3731, pp. 34–37, 1966.

[132] R. S. Sutton, A. Koop, and D. Silver, "On the role of tracking in stationary environments," in Proceedings of the 24th international conference on Machine learning, 2007, pp. 871–878.

[133] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," in 2020 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2020, pp. 737–744.

[134] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," The International Journal of Robotics Research, vol. 32, no. 11, pp. 1238–1274, 2013.

[135] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, J. Peters et al., "An algorithmic perspective on imitation learning," Foundations and Trends® in Robotics, vol. 7, no. 1-2, pp. 1–179, 2018.

[136] B. Nemec, M. Zorko, and L. Žlajpah, "Learning of a ball-in-a-cup playing robot," in 19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD 2010). IEEE, 2010, pp. 297–301.

[137] M. Tokic, W. Ertel, and J. Fessler, "The crawler, a class room demonstrator for reinforcement learning." in FLAIRS Conference, 2009, pp. 2471–2482.

[138] J. Piater, S. Jodogne, R. Detry, D. Kraft, N. Krüger, O. Kroemer, and J. Peters, "Learning visual representations for perception-action systems," The International Journal of Robotics Research, vol. 30, no. 3, pp. 294–307, 2011.

[139] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," ACM Computing Surveys (CSUR), vol. 50, no. 2, pp. 1–35, 2017.

[140] S. Arora and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress," Artificial Intelligence, vol. 297, p. 103500, 2021.

[141] A. Y. Ng, S. J. Russell et al., "Algorithms for inverse reinforcement learning." in Icml, vol. 1, 2000, p. 2.

[142] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in Proceedings of the twenty-first international conference on Machine learning, 2004, p. 1.

[143] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey et al., "Maximum entropy inverse reinforcement learning." in Aaai, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.

[144] J. Ho and S. Ermon, "Generative adversarial imitation learning," Advances in neural information processing systems, vol. 29, pp. 4565–4573, 2016.

[145] Z. Xu, H. Chang, C. Tang, C. Liu, and M. Tomizuka, "Toward modularization of neural network autonomous driving policy using parallel attribute networks," in 2019 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2019, pp. 1400–1407.

[146] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," Neural computation, vol. 3, no. 1, pp. 79–87, 1991.

[147] R. A. Brooks, "Intelligence without representation," Artificial intelligence, vol. 47, no. 1-3, pp. 139–159, 1991.

[148] A. G. Barto and S. Mahadevan, "Recent advances in hierarchical reinforcement learning," Discrete event dynamic systems, vol. 13, no. 1, pp. 41–77, 2003.

[149] O. Nachum, S. S. Gu, H. Lee, and S. Levine, "Data-efficient hierarchical reinforcement learning," in Advances in Neural Information Processing Systems, 2018, pp. 3303–3313.

[150] A. Levy, R. Platt, and K. Saenko, "Hierarchical actor-critic," arXiv preprint arXiv:1712.00948, vol. 12, 2017.

[151] B. Beyret, A. Shafti, and A. A. Faisal, "Dot-to-dot: Explainable hierarchical reinforcement learning for robotic manipulation," arXiv preprint arXiv:1904.06703, 2019.

[152] H. Zhu, J. Yu, A. Gupta, D. Shah, K. Hartikainen, A. Singh, V. Kumar, and S. Levine, "The ingredients of real-world robotic reinforcement learning," arXiv preprint arXiv:2004.12570, 2020.

[153] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow, L. Shamseer, J. M. Tetzlaff, E. A. Akl, S. E. Brennan et al., "The prisma 2020 statement: an updated guideline for reporting systematic reviews," Bmj, vol. 372, 2021.

[154] M. E. Rose and J. R. Kitchin, "pybliometrics: Scriptable bibliometrics using a python interface to scopus," SoftwareX, vol. 10, p. 100263, 2019.

[155] F. E. Vuik et al., "Colon capsule endoscopy in colorectal cancer screening: a systematic review," Endoscopy, vol. 53, no. 08, pp. 815–824, 2021.

[156] K. Meng, Y. Jia, H. Yang, F. Niu, Y. Wang, and D. Sun, "Motion planning and robust control for the endovascular navigation of a microrobot," IEEE Transactions on Industrial Informatics, 2019.

[157] L. Yang, J. Qi, D. Song, J. Xiao, J. Han, and Y. Xia, "Survey of robot 3d path planning algorithms," Journal of Control Science and Engineering, 2016.

[158] B. Geiger, A. P. Kiraly, D. P. Naidich, and C. L. Novak, "Virtual bronchoscopy of peripheral nodules using arteries as surrogate pathways," in Medical Imaging 2005: Physiology, Function, and Structure from Medical Images, A. A. Amini and A. Manduca, Eds. SPIE, 2005.

[159] C. Sánchez, M. Diez-Ferrer, J. Bernal, F. J. Sánchez, A. Rosell, and D. Gil, "Navigation path retrieval from videobronchoscopy using bronchial branches," in Clinical Image-Based Procedures. Translational Research in Medical Imaging. Springer International Publishing, pp. 62–70, 2016.

[160] J. Wang, T. Ohya, H. Liao, I. Sakuma, T. Wang, I. Tohnai, and T. Iwai, "Intravascular catheter navigation using path planning and virtual visual feedback for oral cancer treatment," The International Journal of Medical Robotics and Computer Assisted Surgery, vol. 7, no. 2, pp. 214–224, 2011.

[161] F. Yang, Z.-G. Hou, S.-H. Mi, G.-B. Bian, and X.-L. Xie, "Centerlines extraction for lumen model of human vasculature for computer-aided simulation of intravascular procedures," in Proceeding of the 11th World Congress on Intelligent Control and Automation. IEEE, pp. 970–975, 2014.

[162] M. Kerschnitzki, P. Kollmannsberger, M. Burghammer, G. N. Duda, R. Weinkamer, W. Wagermaier, and P. Fratzl, "Architecture of the osteocyte network correlates with bone material quality," Journal of bone and mineral research, vol. 28, no. 8, pp. 1837–1845, 2013.

[163] W. Yudong et al., "Rapid path extraction and three-dimensional roaming of the virtual endonasal endoscope," Chin. J. Electronics, vol. 30, no. 3, pp. 397–405, 2021.

[164] X. Zang et al., "Image-guided ebus bronchoscopy system for lung-cancer staging," Inform. in medicine unlocked, vol. 25, p. 100665, 2021.

[165] J. D. Gibbs, M. W. Graham, R. Bascom, D. C. Cornish, R. Khare, and W. E. Higgins, "Optimal procedure planning and guidance system for peripheral bronchoscopy," IEEE Transactions on Biomedical Engineering, vol. 61, no. 3, pp. 638–657, 2013.

[166] D. Huang, W. Tang, Y. Ding, T. Wan, and Y. Chen, "An interactive 3d preoperative planning and training system for minimally invasive vascular surgery," in 12th International Conference on Computer-Aided Design and Computer Graphics. IEEE, pp. 443–449, 2011.

[167] C. Fischer et al., "Using magnetic fields to navigate and simultaneously localize catheters in endoluminal environments," IEEE Robot. Automat. Lett., 2022.

[168] S. Schafer, V. Singh, K. R. Hoffmann, P. B. Noël, and J. Xu, "Planning image-guided endovascular interventions: guidewire simulation using shortest path algorithms," in Medical Imaging: Visualization and Image-Guided Procedures. SPIE, 2007.

[169] J. Egger, Z. Mostarkic, S. Grosskopf, and B. Freisleben, "A fast vessel centerline extraction algorithm for catheter simulation," in Twentieth IEEE International Symposium on Computer-Based Medical Systems (CBMS'07), pp. 177-182, 2007.

[170] H. Liu, Y. Fu, Y. Zhou, H. Li, Z. Liang, and S. Wang, "An in vitro investigation of image-guided steerable catheter navigation," Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine, vol. 224, no. 8, pp. 945–954, 2010.

[171] J. D. Gibbs and W. E. Higgins, "3d path planning and extension for endoscopic guidance," in Medical Imaging: Visualization and Image-Guided Procedures. International Society for Optics and Photonics, 2007.

[172] J. D. Gibbs, M. W. Graham, K.-C. Yu, and W. E. Higgins, “Integrated system for planning peripheral bronchoscopic procedures,” in Medical Imaging: Physiology, Function, and Structure from Medical Images. International Society for Optics and Photonics, 2008.

[173] H. Qian, X. Lin, Z. Wu, Q. Zeng, C. Li, Y. Pang, C. Wang, and S. Zhou, “Towards rebuild the interventionist’s intra-operative natural behavior: A fully sensorized endovascular robotic system design,” in International Conference on Medical Imaging Physics and Engineering (ICMIPE). IEEE, pp. 1–7, 2019.

[174] P. Schegg et al., “Automated planning for robotic guidewire navigation in the coronary arteries,” in Proc. IEEE Int. Conf. Soft Robot., 2022, pp. 239–246.

[175] Y. Cho et al., “Image processing based autonomous guidewire navigation in percutaneous coronary intervention,” in Proc. IEEE Int. Conf. Consum. Electron. Asia, 2021.

[176] J. Rosell, A. Pérez, P. Cabras, and A. Rosell, “Motion planning for the virtual bronchoscopy,” in 2012 IEEE International Conference on Robotics and Automation. IEEE, 2012, pp. 2932–2937.

[177] F. Yang, Y. Dai, J. Zhang, H. Sun, L. Cui, X. Yin, X. Gao, and L. Li, “Path planning of flexible ureteroscope based on ct image,” in 2019 Chinese Control Conference (CCC). IEEE, 2019, pp. 4667–4672.

[178] J. W. Martin, B. Scaglioni, J. C. Norton, V. Subramanian, A. Arezzo, K. L. Obstein, and P. Valdastri, “Enabling the future of colonoscopy with intelligent and autonomous magnetic manipulation,” Nature Machine Intelligence, vol. 2, no. 10, pp. 595–606, 2020.

[179] Q. Zhang et al., “Enabling autonomous colonoscopy intervention using a robotic endoscope platform,” IEEE Trans. Biomed. Eng., vol. 68, no. 6, pp. 1957–1968, 2020.

[180] C. Girerd, A. V. Kudryavtsev, P. Rougeot, P. Renaud, K. Rabenorosoa, and B. Tamadazte, “Slam-based follow-the-leader deployment of concentric tube robots,” IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 548–555, 2020.

[181] Y. He, P. Zhang, X. Qi, B. Zhao, S. Li, and Y. Hu, “Endoscopic path planning in robot-assisted endoscopic nasal surgery,” IEEE Access, vol. PP, pp. 1–1, 01 2020.

[182] C. Ciobirca, T. Lango, G. Gruionu, H. O. Leira, L. Gruionu, and S. Pastrama, “A new procedure for automatic path planning in bronchoscopy,” Materials Today: Proceedings, vol. 5, no. 13, pp. 26 513–26 518, 2018.

[183] S. Niyaz, A. Kuntz, O. Salzman, R. Alterovitz, and S. Srinivasa, “Following surgical trajectories with concentric tube robots via nearest-neighbor graphs,” in Int. Symp. Experimental Robotics (ISER), 2018.

[184] S. Niyaz, A. Kuntz, O. Salzman, R. Alterovitz, and S. S. Srinivasa, “Optimizing motion-planning problem setup via bounded evaluation with application to following surgical trajectories,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 1355–1362, 2019.

[185] N. Koenig and A. Howard, “Design and use paradigms for gazebo, an open-source multi-robot simulator,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566), vol. 3. IEEE, pp. 2149–2154, 2004.

[186] S. R. Ravigopal et al., "Automated motion control of the coast robotic guidewire under fluoroscopic guidance," in Proc. Int. Symp. Med. Robot., 2021.

[187] H.-E. Huang et al., "Autonomous navigation of a magnetic colonoscope using force sensing and a heuristic search algorithm," Sci. Rep., vol. 11, no. 1, pp. 1–15, 2021.

[188] I. Cheng et al., "Enhanced segmentation and skeletonization for endovascular surgical planning," in Proc. Med. Imag.: Image-Guided Procedures, Robot. Interv., and Model., vol. 8316, 2012, pp. 868–874.

[189] R. Tarjan, "Depth-first search and linear graph algorithms," SIAM journal on computing, vol. 1, no. 2, pp. 146–160, 1972.

[190] R. Dechter et al., "Generalized best-first search strategies and the optimality of a," J. ACM, vol. 32, no. 3, pp. 505–536, 1985.

[191] E. W. Dijkstra et al., "A note on two problems in connexion with graphs," Numerische mathematik, vol. 1, no. 1, pp. 269–271, 1959.

[192] Y. K. Hwang et al., "A potential field approach to path planning." IEEE Trans. Robot. Automat., vol. 8, no. 1, pp. 23–32, 1992.

[193] P. E. Hart et al., "A formal basis for the heuristic determination of minimum cost paths," IEEE Trans. Syst. Sci. Cybernetics, vol. 4, no. 2, pp. 100–107, 1968.

[194] S. M. LaValle et al., "Rapidly-exploring random trees: A new tool for path planning," 1998.

[195] R. Geraerts et al., "A comparative study of probabilistic roadmap planners," in Algorithmic Found. Robot., 2004, pp. 43–57.

[196] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," The international journal of robotics research, vol. 30, no. 7, pp. 846–894, 2011.

[197] P. Raja et al., "Optimal path planning of mobile robots: A review," Int. J. Phys. Sci., vol. 7, no. 9, pp. 1314–1320, 2012.

[198] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," IEEE computational intelligence magazine, vol. 1, no. 4, pp. 28–39, 2006.

[199] H. Ravichandar et al., "Recent advances in robot learning from demonstration," Annu. Rev. Control, Robot., and Auton. Syst., vol. 3, pp. 297–330, 2020.

[200] G. Fagogenis, M. Mencattelli, Z. Machaidze, B. Rosa, K. Price, F. Wu, V. Weixler, M. Saeed, J. Mayer, and P. Dupont, "Autonomous robotic intracardiac catheter navigation using haptic vision," Science robotics, vol. 4, no. 29, 2019.

[201] W. G. Aguilar, V. Abad, H. Ruiz, J. Aguilar, and F. Aguilar-Castillo, "Rrt-based path planning for virtual bronchoscopy simulator," in International Conference on Augmented Reality, Virtual Reality and Computer Graphics. Springer, pp. 155–165, 2017.

[202] ——, "Virtual bronchoscopy motion planner," in IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON). IEEE, pp. 1–4, 2017.

[203] C. Fellmann and J. Burgner-Kahrs, "Implications of trajectory generation strategies for tubular continuum robots," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 202–208, 2015.

[204] A. Kuntz, L. G. Torres, R. H. Feins, R. J. Webster, and R. Alterovitz, "Motion planning for a three-stage multilumen transoral lung access system," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 3255–3261, 2015.

[205] J. Guo et al., "A training system for vascular interventional surgeons based on local path planning," in Proc. IEEE Int. Conf. Mechatronics and Automat., 2021, pp. 1328–1333.

[206] R. Alterovitz, S. Patil, and A. Derbakova, "Rapidly-exploring roadmaps: Weighing exploration vs. refinement in optimal motion planning," in 2011 IEEE International Conference on Robotics and Automation. IEEE, 2011, pp. 3706–3712.

[207] L. G. Torres and R. Alterovitz, "Motion planning for concentric tube robots using mechanics-based models," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011.

[208] L. G. Torres, R. J. Webster, and R. Alterovitz, "Task-oriented design of concentric tube robots using mechanics-based models," in IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, pp. 4449–4455, 2012.

[209] L. G. Torres, C. Baykal, and R. Alterovitz, "Interactive-rate motion planning for concentric tube robots," in IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 1915–1921, 2014.

[210] J. Fauser, G. Sakas, and A. Mukhopadhyay, "Planning nonlinear access paths for temporal bone surgery," International journal of computer assisted radiology and surgery, vol. 13, no. 5, pp. 637–646, 2018.

[211] J. Fauser, I. Stenin, J. Kristin, T. Klenzner, J. Schipper, D. Fellner, and A. Mukhopadhyay, "Generalized trajectory planning for nonlinear interventions," in OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis. Springer, pp. 46–53, 2018.

[212] J. Fauser, I. Stenin, J. Kristin, T. Klenzner, J. Schipper, and A. Mukhopadhyay, "Optimizing clearance of bézier spline trajectories for minimally-invasive surgery," in International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 20–28, 2019.

[213] J. Fauser, R. Chadda, Y. Goergen, M. Hessinger, P. Motzki, I. Stenin, J. Kristin, T. Klenzner, J. Schipper, S. Seelecke et al., "Planning for flexible surgical robots via bézier spline translation," IEEE Robotics and Automation Letters, vol. 4, no. 4, pp. 3270–3277, 2019.

[214] A. Kuntz, M. Fu, and R. Alterovitz, "Planning high-quality motions for concentric tube robots in point clouds via parallel sampling and optimization," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 2205–2212, 2019.

[215] L. A. Lyons, R. J. Webster, and R. Alterovitz, "Planning active cannula configurations through tubular anatomy," in IEEE international conference on robotics and automation. IEEE, pp. 2082–2087, 2010.

[216] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," Mathematical programming, vol. 45, no. 1, pp. 503–528, 1989.

[217] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, Nonlinear programming: theory and algorithms. John Wiley & Sons, 2013.

[218] F. Qi, F. Ju, D. Bai, Y. Wang, and B. Chen, "Kinematic analysis and navigation method of a cable-driven continuum robot used for minimally invasive surgery," The International Journal of Medical Robotics and Computer Assisted Surgery, p. e2007, 2019.

[219] J. Guo et al., "Design a novel of path planning method for the vascular interventional surgery robot based on dwa model," in Proc. IEEE Int. Conf. Mechatronics and Automat., 2021, pp. 1322–1327.

[220] C. Abah et al., "Image-guided optimization of robotic catheters for patient-specific endovascular intervention," in Proc. IEEE Int. Symp. Med. Robot., 2021, pp. 1–8.

[221] M.-k. Gao, Y.-m. Chen, Q. Liu, C. Huang, Z.-y. Li, and D.-h. Zhang, "Three-dimensional path planning and guidance of leg vascular based on improved ant colony algorithm in augmented reality," Journal of medical systems, vol. 39, no. 11, p. 133, 2015.

[222] Z. Li, J. Dankelman, and E. De Momi, "Path planning for endovascular catheterization under curvature constraints via two-phase searching approach," International Journal of Computer Assisted Radiology and Surgery, vol. 16, no. 4, pp. 619–627, 2021.

[223] H. Rafii-Tari, J. Liu, C. J. Payne, C. Bicknell, and G.-Z. Yang, "Hierarchical hmm based learning of navigation primitives for cooperative robotic endovascular catheterization," in International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 496–503, 2014.

[224] H. Rafii-Tari, J. Liu, S.-L. Lee, C. Bicknell, and G.-Z. Yang, "Learning-based modeling of endovascular navigation for collaborative robotic catheterization," in Advanced Information Systems Engineering. Springer Berlin Heidelberg, pp. 369–377, 2013.

[225] W. Chi et al., "Learning-based endovascular navigation through the use of non-rigid registration for collaborative robotic catheterization," International Journal of Computer Assisted Radiology and Surgery, vol. 13, no. 6, pp. 855–864, Apr. 2018.

[226] M. Saveriano et al., "Dynamic movement primitives in robotics: A tutorial survey," arXiv preprint arXiv:2102.03861, 2021.

[227] W. Chi, G. Dagnino, T. Kwok, A. Nguyen, D. Kundrat, M. E. M. K. Abdelaziz, C. Riga, C. Bicknell, and G.-Z. Yang, "Collaborative robot-assisted endovascular catheterization with generative adversarial imitation learning," in IEEE International Conference on Robotics and Automation (ICRA). IEEE, June 2020.

[228] Y. Zhao et al., "Surgical gan: Towards real-time path planning for passive flexible tools in endovascular surgeries," Neurocomputing, 2022.

[229] F. Meng et al., "Evaluation of a reinforcement learning algorithm for vascular intervention surgery," in Proc. IEEE Int. Conf. Mechatronics and Automat., 2021, pp. 1033–1037.

[230] G. Trovato et al., "Development of a colon endoscope robot that adjusts its locomotion through the use of reinforcement learning," International journal of computer assisted radiology and surgery, vol. 5, no. 4, pp. 317–325, 2010.

[231] T. Behr, T. P. Pusch, M. Siegfarth, D. Hüsener, T. Mörschel, and L. Karstensen, "Deep reinforcement learning for the navigation of neurovascular catheters," Current Directions in Biomedical Engineering, vol. 5, no. 1, pp. 5–8, 2019.

[232] L. Karstensen, T. Behr, T. P. Pusch, F. Mathis-Ullrich, and J. Stallkamp, "Autonomous guidewire navigation in a two dimensional vascular phantom," Current Directions in Biomedical Engineering, vol. 6, no. 1, 2020.

[233] J. Kweon et al., "Deep reinforcement learning for guidewire navigation in coronary artery phantom," IEEE Access, vol. 9, pp. 166 409–166 422, 2021.

[234] A. Pore, M. Finocchiaro, D. Dall'Alba, A. Hernansanz, G. Ciuti, A. Arezzo, A. Menciassi, A. Casals, and P. Fiorini, "Colonoscopy navigation using end-to-end deep visuomotor control: A user study," in 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2022, pp. 9582–9588.

[235] L. Karstensen et al., "Learning-based autonomous vascular guidewire navigation without human demonstration in the venous system of a porcine liver," Int. J. Comput. Assist. Radiol. Surg., pp. 1–8, 2022.

[236] D. Corsi, L. Marzari, A. Pore, A. Farinelli, A. Casals, P. Fiorini, and D. Dall'Alba, "Constrained reinforcement learning and formal verification for safe colonoscopy navigation," arXiv preprint arXiv:2303.03207, 2023.

[237] J. Zhang, Y. Zhong, and C. Gu, "Deformable models for surgical simulation: a survey," IEEE reviews in biomedical engineering, vol. 11, pp. 143–164, 2017.

[238] S. Li, J. Qin, J. Guo, Y.-P. Chui, and P.-A. Heng, "A novel fem-based numerical solver for interactive catheter simulation in virtual catheterization," International journal of biomedical imaging, vol. 2011, 2011.

[239] I. Badash, K. Burtt, C. A. Solorzano, and J. N. Carey, "Innovations in surgery simulation: a review of past, current and future techniques," Annals of translational medicine, vol. 4, no. 23, 2016.

[240] Y. Wang, F. Serracino-Inglott, X. Yi, X.-F. Yuan, and X.-J. Yang, "Real-time simulation of catheterization in endovascular surgeries," Computer Animation and Virtual Worlds, vol. 27, no. 3-4, pp. 185–194, 2016.

[241] V. Luboz, J. Zhai, T. Odetoyinbo, P. Littler, D. Gould, T. How, and F. Bello, "Simulation of endovascular guidewire behaviour and experimental validation," Computer methods in biomechanics and biomedical engineering, vol. 14, no. 06, pp. 515–520, 2011.

[242] V. Guilloux, P. Haigron, C. Goksu, C. Kulik, and A. Lucas, "Simulation of guide-wire navigation in complex vascular structures," in Medical Imaging 2006: Visualization, Image-Guided Procedures, and Display, vol. 6141. International Society for Optics and Photonics, 2006, p. 614107.

[243] K. TAKASHIMA, A. OIKE, K. YOSHINAKA, K. YU, M. OHTA, K. MORI, and N. TOMA, "Evaluation of the effect of catheter on the guidewire motion in a blood vessel model by physical and numerical simulations," Journal of Biomechanical Science and Engineering, vol. 12, no. 4, pp. 17–00 181, 2017.

[244] W. Tang, T. R. Wan, D. A. Gould, T. How, and N. W. John, "A stable and real-time nonlinear elastic approach to simulating guidewire and catheter insertions based on cosserat rod," IEEE Transactions on Biomedical Engineering, vol. 59, no. 8, pp. 2211–2218, 2012.

[245] S.-H. Mi, Z.-G. Hou, F. Yang, X.-L. Xie, and G.-B. Bian, "A multi-body mass-spring model for virtual reality training simulators based on a robotic guide wire operating system," in 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO). IEEE, 2013, pp. 2031–2036.

[246] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka et al., "3d slicer as an image computing platform for the quantitative imaging network," Magnetic resonance imaging, vol. 30, no. 9, pp. 1323–1341, 2012.

[247] L. Antiga and D. A. Steinman, "Vmtk: vascular modeling toolkit," VMTK, San Francisco, CA, accessed Apr, vol. 27, p. 2015, 2006.

[248] A. P. Kiraly, J. P. Helferty, E. A. Hoffman, G. McLennan, and W. E. Higgins, "Three-dimensional path planning for virtual bronchoscopy," IEEE Transactions on Medical Imaging, vol. 23, no. 11, pp. 1365–1379, 2004.

[249] A. Updegrove, N. M. Wilson, J. Merkow, H. Lan, A. L. Marsden, and S. C. Shadden, "Simvascular: an open source pipeline for cardiovascular simulation," Annals of biomedical engineering, vol. 45, no. 3, pp. 525–541, 2017.

[250] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik et al., "Sofa: A multi-model framework for interactive physical simulation," in Soft tissue biomechanical modeling for computer assisted surgery. Springer, 2012, pp. 283–321.

[251] A. Bihlmaier and H. Woern, "Automated endoscopic camera guidance: A knowledge-based system towards robot assisted surgery," in ISR/Robotik 2014; 41st International Symposium on Robotics. VDE, 2014, pp. 1–6.

[252] S. Yi, H. Woo, W. Ahn, J. Kwon, and D. Lee, "New colonoscopy simulator with improved haptic fidelity," Advanced Robotics, vol. 20, no. 3, pp. 349–365, 2006.

[253] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff, "Position based dynamics," Journal of Visual Communication and Image Representation, vol. 18, no. 2, pp. 109–118, 2007.

[254] D. Zhou and X. He, "Numerical evaluation of the efficacy of small-caliber colonoscopes in reducing patient pain during a colonoscopy," Computer methods in biomechanics and biomedical engineering, vol. 22, no. 1, pp. 38–46, 2019.

[255] K. Ratnayaka, T. Rogers, W. H. Schenke, J. R. Mazal, M. Y. Chen, M. Sonmez, M. S. Hansen, O. Kocaturk, A. Z. Faranesh, and R. J. Lederman, "Magnetic resonance imaging–guided transcatheter cavopulmonary shunt," JACC: Cardiovascular Interventions, vol. 9, no. 9, pp. 959–970, May 2016.

[256] P. Dupont, N. Simaan, H. Choset, and C. Rucker, "Continuum robots for medical interventions," Proceedings of the IEEE, 2022.

[257] F. Liu, A. Garriga-Casanovas, R. Secoli, and F. R. y Baena, "Fast and adaptive fractal tree-based path planning for programmable bevel tip steerable needles," IEEE Robotics and Automation Letters, vol. 1, no. 2, pp. 601–608, 2016.

[258] M. Pinzi, S. Galvan, and F. R. y Baena, "The adaptive hermite fractal tree (ahft): a novel surgical 3d path planning approach with curvature and heading constraints," International journal of computer assisted radiology and surgery, vol. 14, no. 4, pp. 659–670, 2019.

[259] W. Wang, L. Zuo, and X. Xu, "A learning-based multi-rrt approach for robot path planning in narrow passages," Journal of Intelligent & Robotic Systems, vol. 90, no. 1-2, pp. 81–100, 2018.

[260] J. Granna, A. Nabavi, and J. Burgner-Kahrs, "Computer-assisted planning for a concentric tube robotic system in neurosurgery," International journal of computer assisted radiology and surgery, vol. 14, no. 2, pp. 335–344, 2019.

[261] M. J. Pourmanda and M. Sharifib, "Navigation and control of endovascular helical swimming microrobot using dynamic programing and adaptive sliding mode strategy," Control Systems Design of Bio-Robotics and Bio-Mechatronics with Advanced Applications, p. 201, 2019.

[262] T. A. Howell, B. E. Jackson, and Z. Manchester, "Altro: A fast solver for constrained trajectory optimization," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 7674–7679, 2019.

[263] D. C. Birkhoff et al., "A review on the current applications of artificial intelligence in the operating room," Surg. Innov., vol. 28, no. 5, pp. 611–619, 2021.

[264] M. G. Goldenberg et al., "Using data to enhance performance and improve quality and safety in surgery," JAMA Surg., vol. 152, no. 10, pp. 972–973, 2017.

[265] J. Garcıa and F. Fernández, "A comprehensive survey on safe reinforcement learning," Journal of Machine Learning Research, vol. 16, no. 1, pp. 1437–1480, 2015.

[266] S. K. S. Ghasemipour et al., "A divergence minimization perspective on imitation learning methods," in Proc. PMLR Conf. Robot. Learn., 2020, pp. 1259–1277.

[267] L. Ke et al., "Imitation learning as f-divergence minimization," in Int. Workshop on the Algorithmic Found. Robot., 2020, pp. 313–329.

[268] I. A. Seleem, H. El-Hussieny, S. F. Assal, and H. Ishii, "Development and stability analysis of an imitation learning-based pose planning approach for multi-section continuum robot," IEEE Access, 2020.

[269] H. Sung, J. Ferlay, R. L. Siegel, M. Laversanne, I. Soerjomataram, A. Jemal, and F. Bray, "Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries," CA: a cancer journal for clinicians, vol. 71, no. 3, pp. 209–249, 2021.

[270] Observatory (2021), Global Cancer Observatory, 2021, available at https://gco.iarc.fr/.

[271] C. Chen, M. Hoffmeister, and H. Brenner, "The toll of not screening for colorectal cancer," Expert review of gastroenterology & hepatology, vol. 11, no. 1, pp. 1–3, 2017.

[272] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, "Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries," CA: a cancer journal for clinicians, vol. 68, no. 6, pp. 394–424, 2018.

[273] A. K. Shergill, K. R. McQuaid, and D. Rempel, "Ergonomics and gi endoscopy," Gastrointestinal endoscopy, vol. 70, no. 1, pp. 145–153, 2009.

[274] S.-H. Lee, Y.-K. Park, D.-J. Lee, and K.-M. Kim, "Colonoscopy procedural skills and training for new beginners," World Journal of Gastroenterology: WJG, vol. 20, no. 45, p. 16984, 2014.

[275] G. Iddan, G. Meron, A. Glukhovsky, and P. Swain, "Wireless capsule endoscopy," Nature, vol. 405, no. 6785, pp. 417–417, 2000.

[276] F. Nageotte, L. Zorn, P. Zanne, and M. De Mathelin, "Stras: A modular and flexible telemanipulated robotic device for intraluminal surgery," in Handbook of Robotic and Image-Guided Surgery. Elsevier, 2020, pp. 123–146.

[277] A. Z. Taddese, P. R. Slawinski, K. L. Obstein, and P. Valdastri, "Nonholonomic closed-loop velocity control of a soft-tethered magnetic capsule endoscope," in 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2016, pp. 1139–1144.

[278] D. Kragic, H. I. Christensen et al., "Survey on visual servoing for manipulation," Computational Vision and Active Perception Laboratory, Fiskartorpsv, vol. 15, p. 2002, 2002.

[279] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. d. L. Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq et al., "Deepmind control suite," arXiv preprint arXiv:1801.00690, 2018.

[280] A. Saxena, H. Pandya, G. Kumar, A. Gaud, and K. M. Krishna, "Exploring convolutional networks for end-to-end visual servoing," in 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017, pp. 3817–3823.

[281] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," The Journal of Machine Learning Research, vol. 17, no. 1, pp. 1334–1373, 2016.

[282] J. Xu, B. Li, B. Lu, Y.-H. Liu, Q. Dou, and P.-A. Heng, "Surrol: An open-source reinforcement learning centered and dvrk compatible platform for surgical robot learning," in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2021, pp. 1821–1828.

[283] P. M. Scheikl, B. Gyenes, T. Davitashvili, R. Younis, A. Schulze, B. P. Müller-Stich, G. Neumann, M. Wagner, and F. Mathis-Ullrich, "Cooperative assistance in robotic surgery through multi-agent reinforcement learning," in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2021, pp. 1859–1864.

[284] Y.-H. Su, K. Huang, and B. Hannaford, "Multicamera 3d viewpoint adjustment for robotic surgery via deep reinforcement learning," Journal of Medical Robotics Research, vol. 6, no. 01n02, p. 2140003, 2021.

[285] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle et al., "The cancer imaging archive (tcia): maintaining and operating a public information repository," Journal of digital imaging, vol. 26, no. 6, pp. 1045–1057, 2013.

[286] B. H. Jeong, H. K. Kim, and Y. D. Son, "Depth estimation of endoscopy using sim-to-real transfer," arXiv preprint arXiv:2112.13595, 2021.

[287] K. Pogorelov, K. R. Randel, C. Griwodz, S. L. Eskeland, T. de Lange, D. Johansen, C. Spampinato, D.-T. Dang-Nguyen, M. Lux, P. T. Schmidt et al., "Kvasir: A multi-class image dataset for computer aided gastrointestinal disease detection," in Proceedings of the 8th ACM on Multimedia Systems Conference, 2017, pp. 164–169.

[288] M. B. Christensen, K. Oberg, and J. C. Wolchok, "Tensile properties of the rectal and sigmoid colon: a comparative analysis of human and porcine tissue," Springerplus, vol. 4, no. 1, pp. 1–10, 2015.

[289] D. Wang, X. Xie, G. Li, Z. Yin, and Z. Wang, "A lumen detection-based intestinal direction vector acquisition method for wireless endoscopy systems," IEEE Transactions on Biomedical Engineering, vol. 62, no. 3, pp. 807–819, 2014.

[290] M. Merino-Monge, A. J. Molina-Cantero, J. A. Castro-García, and I. M. Gómez-González, "An easy-to-use multi-source recording and synchronization software for experimental trials," IEEE Access, vol. 8, pp. 200 618–200 634, 2020.

[291] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in Advances in psychology. Elsevier, 1988, vol. 52, pp. 139–183.

[292] M. F. Kaminski, S. Thomas-Gibson, M. Bugajski, M. Bretthauer, C. J. Rees, E. Dekker, G. Hoff, R. Jover, S. Suchanek, M. Ferlitsch et al., "Performance measures for lower gastrointestinal endoscopy: a european society of gastrointestinal endoscopy (esge) quality improvement initiative," Endoscopy, vol. 49, no. 04, pp. 378–397, 2017.

[293] N. van der Stap, C. H. Slump, I. A. Broeders, and F. van der Heijden, "Image-based navigation for a robotized flexible endoscope," in Computer-Assisted and Robotic Endoscopy: First International Workshop, CARE 2014, Held in Conjunction with MICCAI 2014, Boston, MA, USA, September 18, 2014. Revised Selected Papers 1. Springer, 2014, pp. 77–87.

[294] J. M. Prendergast, G. A. Formosa, M. J. Fulton, C. R. Heckman, and M. E. Rentschler, "A real-time state dependent region estimator for autonomous endoscope navigation," IEEE Transactions on Robotics, vol. 37, no. 3, pp. 918–934, 2020.

[295] R. Reilink, S. Stramigioli, and S. Misra, "Image-based flexible endoscope steering," in 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2010, pp. 2339–2344.

[296] J. F. Lazo, C.-F. Lait, S. Moccia, B. Rosa, M. Catellani, M. de Mathelin, G. Ferrigno, P. Breedveld, J. Dankelman, and E. De Momi, "Autonomous intraluminal navigation of a soft robot using deep-learning-based visual servoing," in 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2022, pp. 6952–6959.

[297] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in International Conference on Learning Representations, 2015.

[298] J. Garcıa and F. Fernández, “A comprehensive survey on safe reinforcement learning,” Journal of Machine Learning Research, 2015.

[299] Y. Liu, J. Ding, and X. Liu, “Ipo: Interior-point policy optimization under constraints,” in Proceedings of the AAAI Conference on Artificial Intelligence, 2020.

[300] J. Achiam et al., “Constrained policy optimization,” in Int. Conf. Mach. Learn., 2017.

[301] E. Marchesini, D. Corsi, and A. Farinelli, “Exploring Safer Behaviors for Deep Reinforcement Learning,” in Proc. 35th AAAI Conf. on Artificial Intelligence (AAAI), 2021.

[302] A. Ray, J. Achiam, and D. Amodei, “Benchmarking safe exploration in deep reinforcement learning,” arXiv preprint arXiv:1910.01708, 2019.

[303] D. Yu, H. Ma, S. Li, and J. Chen, “Reachability constrained reinforcement learning,” in International Conference on Machine Learning. PMLR, 2022, pp. 25 636–25 655.

[304] C. Liu, T. Arnon, C. Lazarus, C. Strong, C. Barrett, M. J. Kochenderfer et al., “Algorithms for verifying deep neural networks,” Foundations and Trends® in Optimization, vol. 4, no. 3-4, pp. 244–404, 2021.

[305] G. Katz et al., “Reluplex: An efficient smt solver for verifying deep neural networks,” in Int. conf. comp. verif. Springer, 2017, pp. 97–117.

[306] L. Marzari, D. Corsi, F. Cicalese, and A. Farinelli, “The# dnn-verification problem: Counting unsafe inputs for deep neural networks,” arXiv preprint arXiv:2301.07068, 2023.

[307] D. Corsi et al., “Formal verification of neural networks for safety-critical tasks in deep reinforcement learning,” in Uncert. Artif. Intel. PMLR, 2021, pp. 333–343.

[308] G. Amir, D. Corsi, R. Yerushalmi, L. Marzari, D. Harel, A. Farinelli, and G. Katz, “Verifying learning-based robotic navigation systems,” arXiv preprint arXiv:2205.13536, 2022.

[309] A. Pore, D. Corsi, E. Marchesini, D. Dall’Alba, A. Casals, A. Farinelli, and P. Fiorini, “Safe reinforcement learning using formal verification for tissue retraction in autonomous robotic-assisted surgery,” in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2021, pp. 4025–4031.

[310] P. Henriksen and A. Lomuscio, “Deepsplit: An efficient splitting method for neural network verification via indirect effect analysis.” in IJCAI, 2021, pp. 2549–2555.

[311] M. Than, J. Witherspoon, J. Shami, P. Patil, and A. Saklani, “Diagnostic miss rate for colorectal cancer: an audit,” Annals of gastroenterology: quarterly publication of the Hellenic Society of Gastroenterology, vol. 28, no. 1, p. 94, 2015.

[312] S. Menon and N. Trudgill, “How commonly is upper gastrointestinal cancer missed at endoscopy? a meta-analysis,” Endoscopy international open, vol. 2, no. 02, pp. E46–E50, 2014.

[313] L. Xiao, X. Yu, W. Deng, H. Feng, H. Chang, W. Xiao, H. Zhang, S. Xi, M. Liu, Y. Zhu et al., “Pathological assessment of rectal cancer after neoadjuvant chemoradiotherapy: distribution of residual cancer cells and accuracy of biopsy,” Scientific reports, vol. 6, no. 1, pp. 1–7, 2016.

[314] A. De Donno, L. Zorn, P. Zanne, F. Nageotte, and M. de Mathelin, "Introducing stras: A new flexible robotic system for minimally invasive surgery," in 2013 IEEE International Conference on Robotics and Automation. IEEE, 2013, pp. 1213–1220.

[315] O. C. Mora, P. Zanne, L. Zorn, F. Nageotte, N. Zulina, S. Gravelyn, P. Montgomery, M. De Mathelin, B. Dallemagne, and M. J. Gora, "Steerable oct catheter for real-time assistance during teleoperated endoscopic treatment of colorectal cancer," Biomedical optics express, vol. 11, no. 3, pp. 1231–1243, 2020.

[316] N. Zulina, O. Caravaca, G. Liao, S. Gravelyn, M. Schmitt, K. Badu, L. Heroin, and M. J. Gora, "Colon phantoms with cancer lesions for endoscopic characterization with optical coherence tomography," Biomedical optics express, vol. 12, no. 2, pp. 955–968, 2021.

[317] J. W. Martin, L. Barducci, B. Scaglioni, J. C. Norton, C. Winters, V. Subramanian, A. Arezzo, K. L. Obstein, and P. Valdastri, "Robotic autonomy for magnetic endoscope biopsy," IEEE Transactions on Medical Robotics and Bionics, 2022.

[318] Z. Zhang, B. Rosa, O. Caravaca-Mora, P. Zanne, M. J. Gora, and F. Nageotte, "Image-guided control of an endoscopic robot for oct path scanning," IEEE Robotics and Automation Letters, vol. 6, no. 3, pp. 5881–5888, 2021.

[319] O. Caravaca-Mora, P. Zanne, G. Liao, N. Zulina, L. Heroin, L. Zorn, M. De Mathelin, B. Rosa, F. P. Nageotte, and M. J. Gora, "Automatic intraluminal scanning with a steerable endoscopic optical coherence tomography catheter for gastroenterology applications," Journal of Optical Microsystems, vol. 3, no. 1, p. 011005, 2023.

[320] M. R. Hee, J. A. Izatt, E. A. Swanson, D. Huang, J. S. Schuman, C. P. Lin, C. A. Puliafito, and J. G. Fujimoto, "Optical coherence tomography of the human retina," Archives of ophthalmology, vol. 113, no. 3, pp. 325–332, 1995.

[321] Y. Baran, K. Rabenorosoa, G. J. Laurent, P. Rougeot, N. Andreff, and B. Tamadazte, "Preliminary results on oct-based position control of a concentric tube robot," in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017, pp. 3000–3005.

[322] M. Ourak, B. Tamadazte, and N. Andreff, "Partitioned camera-oct based 6 dof visual servoing for automatic repetitive optical biopsies," in 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2016, pp. 2337–2342.

[323] L.-A. Duflot, B. Tamadazte, N. Andreff, and A. Krupa, "Wavelet-based visual servoing using oct images," in 2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob). IEEE, 2018, pp. 621–626.

[324] Y. Li, Z. Zhu, J. J. Chen, J. C. Jing, C.-H. Sun, S. Kim, P.-S. Chung, and Z. Chen, "Multimodal endoscopy for colorectal cancer detection by optical coherence tomography and near-infrared fluorescence imaging," Biomedical Optics Express, vol. 10, no. 5, pp. 2419–2429, 2019.

[325] J. Mavadia-Shukla, P. Fathi, W. Liang, S. Wu, C. Sears, and X. Li, "High-speed, ultrahigh-resolution distal scanning oct endoscopy at 800 nm for in vivo imaging of colon tumorigenesis on murine models," Biomedical optics express, vol. 9, no. 8, pp. 3731–3739, 2018.

[326] F. Nageotte, L. Zorn, P. Zanne, and M. De Mathelin, “Stras: A modular and flexible telemanipulated robotic device for intraluminal surgery,” in Handbook of Robotic and Image-Guided Surgery. Elsevier, 2020, pp. 123–146.

[327] S. Diamond and S. Boyd, “CVXPY: A Python-embedded modeling language for convex optimization,” Journal of Machine Learning Research, vol. 17, no. 83, pp. 1–5, 2016.

[328] Ronneberger Olaf et al., “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical image computing and computer-assisted intervention. Springer, 2015, pp. 234–241.

[329] G. Liao, O. Caravaca-Mora, B. Rosa, P. Zanne, A. Asch, D. Dall’Alba, P. Fiorini, M. de Mathelin, F. Nageotte, and M. J. Gora, “Data stream stabilization for optical coherence tomography volumetric scanning,” IEEE Transactions on Medical Robotics and Bionics, 2021.

[330] G. Liao, O. Caravaca-Mora, B. Rosa, P. Zanne, D. Dall’Alba, P. Fiorini, M. de Mathelin, F. Nageotte, and M. J. Gora, “Distortion and instability compensation with deep learning for rotational scanning endoscopic optical coherence tomography,” Medical Image Analysis, vol. 77, p. 102355, 2022.

[331] M.-H. Laves, L. A. Kahrs, and T. Ortmaier, “Volumetric 3d stitching of optical coherence tomography volumes,” Current Directions in Biomedical Engineering, vol. 4, no. 1, pp. 327–330, 2018.

[332] G. Liao, B. B. F. Barata, O. C. Mora, P. Zanne, B. Rosa, D. Dall’Alba, P. Fiorini, M. de Mathelin, F. Nageotte, and M. J. Gora, “Coordinates encoding networks: an image segmentation architecture for side-viewing catheters,” in Endoscopic Microscopy XVII. SPIE, 2022.

[333] J. T. Maple, B. K. A. Dayyeh, S. S. Chauhan, J. H. Hwang, S. Komanduri, M. Manfredi, V. Konda, F. M. Murad, U. D. Siddiqui, and S. Banerjee, “Endoscopic submucosal dissection,” Gastrointestinal endoscopy, vol. 81, no. 6, pp. 1311–1325, 2015.

[334] R. Mann, M. Gajendran, C. Umapathy, A. Perisetti, H. Goyal, S. Saligram, and J. Echavarria, “Endoscopic management of complex colorectal polyps: current insights and future trends,” Frontiers in Medicine, vol. 8, p. 3081, 2022.

[335] M. Fujiya, K. Tanaka, T. Dokoshi, M. Tominaga, N. Ueno, Y. Inaba, T. Ito, K. Moriichi, and Y. Kohgo, “Efficacy and adverse events of emr and endoscopic submucosal dissection for the treatment of colon neoplasms: a meta-analysis of studies comparing emr and endoscopic submucosal dissection,” Gastrointestinal endoscopy, vol. 81, no. 3, pp. 583–595, 2015.

[336] J. Ferlay, M. Colombet, I. Soerjomataram, C. Mathers, D. M. Parkin, M. Piñeros, A. Znaor, and F. Bray, “Estimating the global cancer incidence and mortality in 2018: Globocan sources and methods,” International journal of cancer, vol. 144, no. 8, pp. 1941–1953, 2019.

[337] D. Lomanto, S. Wijerathne, L. K. Y. Ho, and L. S. J. Phee, “Flexible endoscopic robot,” Minimally Invasive Therapy & Allied Technologies, vol. 24, no. 1, pp. 37–44, 2015.

[338] C. Warren, A. Hamilton, and A. Stevenson, “Robotic transanal minimally invasive surgery (tamis) for local excision of rectal lesions with the da vinci xi (dvxi): technical considerations and video vignette,” Techniques in Coloproctology, vol. 22, no. 7, pp. 529–533, 2018.

[339] P. R. Steele, J. Curran, and R. Mountain, "Current and future practices in surgical retraction," the surgeon, vol. 11, no. 6, pp. 330–337, 2013.

[340] T. D. Nagy, M. Takács, I. J. Rudas, and T. Haidegger, "Surgical subtask automation?soft tissue retraction," in 2018 IEEE 16th World Symposium on Applied Machine Intelligence and Informatics (SAMI). IEEE, 2018, pp. 000 055–000 060.

[341] S. Patil and R. Alterovitz, "Toward automated tissue retraction in robot-assisted surgery," in 2010 IEEE International Conference on Robotics and Automation. IEEE, 2010, pp. 2088–2094.

[342] R. Jansen, K. Hauser, N. Chentanez, F. Van Der Stappen, and K. Goldberg, "Surgical retraction of non-uniform deformable layers of tissue: 2d robot grasping and path planning," in 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2009, pp. 4092–4097.

[343] J. Schulman, A. Gupta, S. Venkatesan, M. Tayson-Frederick, and P. Abbeel, "A case study of trajectory transfer through non-rigid registration for a simplified suturing scenario," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2013, pp. 4111–4117.

[344] T. Osa, K. Harada, N. Sugita, and M. Mitsuishi, "Trajectory planning under different initial conditions for surgical task automation by learning from demonstration," in 2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2014, pp. 6507–6513.

[345] H. Su, A. Mariani, S. E. Ovur, A. Menciassi, G. Ferrigno, and E. De Momi, "Toward teaching by demonstration for robot-assisted minimally invasive surgery," IEEE Transactions on Automation Science and Engineering, 2021.

[346] C. E. Reiley, E. Plaku, and G. D. Hager, "Motion generation of robotic surgical tasks: Learning from expert demonstrations," in 2010 Annual international conference of the IEEE engineering in medicine and biology. IEEE, 2010, pp. 967–970.

[347] N. D. Nguyen, T. Nguyen, S. Nahavandi, A. Bhatti, and G. Guest, "Manipulating soft tissues by deep reinforcement learning for autonomous robotic surgery," in 2019 IEEE International Systems Conference (SysCon). IEEE, 2019, pp. 1–7.

[348] S. A. Pedram, P. W. Ferguson, C. Shin, A. Mehta, E. P. Dutson, F. Alambeigi, and J. Rosen, "Toward synergic learning for autonomous manipulation of deformable tissues via surgical robots: An approximate q-learning approach," arXiv preprint arXiv:1910.03398, 2019.

[349] P. M. Scheikl, E. Tagliabue, B. Gyenes, M. Wagner, D. Dall'Alba, P. Fiorini, and F. Mathis-Ullrich, "Sim-to-real transfer for visual reinforcement learning of deformable object manipulation for robot-assisted surgery," IEEE Robotics and Automation Letters, 2022.

[350] A. Nair, B. McGrew, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Overcoming exploration in reinforcement learning with demonstrations," in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 6292–6299.

[351] W. Chi, G. Dagnino, T. M. Kwok, A. Nguyen, D. Kundrat, M. E. Abdelaziz, C. Riga, C. Bicknell, and G.-Z. Yang, "Collaborative robot-assisted endovascular catheterization with generative adversarial imitation learning," in 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 2414–2420.

[352] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio, “An open-source research kit for the da vinci® surgical system,” in 2014 IEEE international conference on robotics and automation (ICRA). IEEE, 2014, pp. 6434–6439.

[353] E. Tagliabue, D. Dall’Alba, E. Magnabosco, C. Tenga, I. Peterlik, and P. Fiorini, “Position-based modeling of lesion displacement in ultrasound-guided breast biopsy.” IJCARS, 2019.

[354] A. Juliani, V.-P. Berges, E. Vckay, Y. Gao, H. Henry, M. Mattar, and D. Lange, “Unity: A general platform for intelligent agents,” arXiv preprint arXiv:1809.02627, 2018.

[355] (2018) NVIDIA gameworks. Nvidia FleX. [Online]. Available: https://developer.nvidia.com/flex

[356] L. Qian, A. Deguet, and P. Kazanzides, “dVRK-XR: Mixed Reality Extension for da Vinci Research Kit,” in Hamlyn Symposium on Medical Robotics, 2019.

[357] A. Nandy and M. Biswas, “Unity ml-agents,” in Neural Networks in Unity. Springer, 2018, pp. 27–67.

[358] Y. Li, F. Richter, J. Lu, E. K. Funk, R. K. Orosco, J. Zhu, and M. C. Yip, “Super: A surgical perception framework for endoscopic tissue manipulation with surgical robotics,” IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 2294–2301, 2020.

[359] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick, “End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 3387–3395, 2019.

[360] C. Liu, T. Arnon, C. Lazarus, C. Barrett, and M. J. Kochenderfer, “Algorithms for verifying deep neural networks,” in Foundations and Trends in Optimization, 2019.

[361] S. Gu, L. Yang, Y. Du, G. Chen, F. Walter, J. Wang, Y. Yang, and A. Knoll, “A review of safe reinforcement learning: Methods, theory and applications,” arXiv preprint arXiv:2205.10330, 2022.

[362] R. Yang, X. Sun, and K. Narasimhan, “A generalized algorithm for multi-objective reinforcement learning and policy adaptation,” in NeurIPS, 2019.

[363] P. Vamplew, R. Dazeley, A. Berry, R. Issabekov, and E. Dekker, “Empirical evaluation methods for multiobjective reinforcement learning algorithms,” in Machine Learning, 2011.

[364] A. Ray, J. Achiam, and D. Amodei, “Benchmarking safe exploration in deep reinforcement learning,” in OpenAI, 2019.

[365] A. Sinha, H. Namkoong, and J. Duchi, “Certifying some distributional robustness with principled adversarial training,” 2018.

[366] R. E. Moore, “Interval arithmetic and automatic error analysis in digital computing,” in Stanford University, 1963.

[367] L. Weng, H. Zhang, H. Chen, Z. Song, C.-J. Hsieh, L. Daniel, D. Boning, and I. Dhillon, “Towards fast computation of certified robustness for relu networks,” in International Conference on Machine Learning, 2018.

[368] S. Wang, K. Pei, J. Whitehouse, J. Yang, and S. Jana, "Efficient formal safety analysis of neural networks," in Conference on Neural Information Processing Systems, 2018.

[369] S. Fujimoto, H. van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in ICML, 2018.

[370] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel et al., "Soft actor-critic algorithms and applications," arXiv preprint arXiv:1812.05905, 2018.

[371] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," in arXiv, 2017.

[372] R. E. Moore, Interval arithmetic and automatic error analysis in digital computing. Stanford University, 1963.

[373] S. Kolachalama and S. Lakshmanan, "Continuum robots for manipulation applications: A survey," Journal of Robotics, 2020.

[374] A. A. Transeth, K. Y. Pettersen, and P. Liljebäck, "A survey on snake robot modeling and locomotion," Robotica, vol. 27, no. 7, pp. 999–1015, 2009.

[375] Y. Chen, Z. Li, W. Xu, Y. Wang, and H. Ren, "Minimum sweeping area motion planning for flexible serpentine surgical manipulator with kinematic constraints," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 6348–6353, 2015.

[376] J. Fras, M. Macias, Y. Noh, and K. Althoefer, "Fluidical bending actuator designed for soft octopus robot tentacle," in IEEE International Conference on Soft Robotics (RoboSoft). IEEE, pp. 253–257, 2018.

[377] R. Luo, T. Wang, Z. Shi, and J. Tian, "Design and kinematic analysis of an elephant-trunk-like robot with shape memory alloy actuators," in IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). IEEE, pp. 157–161, 2017.

[378] Y. Hu, L. Zhang, W. Li, and G.-Z. Yang, "Design and fabrication of a 3-d printed metallic flexible joint for snake-like surgical robot," IEEE Robotics and Automation Letters, vol. 4, no. 2, pp. 1557–1563, 2019.

[379] M. Neumann and J. Burgner-Kahrs, "Considerations for follow-the-leader motion of extensible tendon-driven continuum robots," in IEEE international conference on robotics and automation (ICRA). IEEE, pp. 917–923, 2016.

[380] L. Pfotzer, S. Klemm, A. Rönnau, J. M. Zöllner, and R. Dillmann, "Autonomous navigation for reconfigurable snake-like robots in challenging, unknown environments," Robotics and Autonomous Systems, vol. 89, pp. 123–135, 2017.

[381] M. Hannan and I. Walker, "The'elephant trunk'manipulator, design and implementation," in IEEE/ASME International Conference on Advanced Intelligent Mechatronics, vol. 1. IEEE, pp. 14–19, 2001.

[382] 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, 2017.

[383] C. Shi, X. Luo, P. Qi, T. Li, S. Song, Z. Najdovski, T. Fukuda, and H. Ren, “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.

[384] X. T. Ha, M. Ourak, O. Al-Ahmad, D. Wu, G. Borghesan, A. Menciassi, and E. Vander Poorten, “Robust catheter tracking by fusing electromagnetic tracking, fiber bragg grating and sparse fluoroscopic images,” IEEE Sensors Journal, vol. 21, no. 20, pp. 23 422–23 434, 2021.

[385] X. T. Ha et al., “Contact localization of continuum and flexible robot using data-driven approach,” IEEE Robot. Automat. Lett., 2022.

[386] L. Sestini et al., “A kinematic bottleneck approach for pose regression of flexible surgical instruments directly from images,” IEEE Robot. Automat. Lett., vol. 6, no. 2, pp. 2938–2945, 2021.

[387] B. F. Barata et al., “Ivus-based local vessel estimation for robotic intravascular navigation,” IEEE Robot. Automat. Lett., vol. 6, no. 4, pp. 8102–8109, 2021.

[388] A. Rau et al., “Implicit domain adaptation with conditional generative adversarial networks for depth prediction in endoscopy,” Int. J. Comput. Assist. Radiol. Surg., vol. 14, no. 7, pp. 1167–1176, 2019.

[389] A. A. Rusu, M. Vecerik, T. Rothörl, N. Heess, R. Pascanu, and R. Hadsell, “Sim-to-real robot learning from pixels with progressive nets,” arXiv preprint arXiv:1610.04286, 2016.

[390] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, “Sim-to-real transfer of robotic control with dynamics randomization,” in IEEE international conference on robotics and automation (ICRA). IEEE, pp. 1–8, 2018.

[391] E. TAGLIABUE, “Patient-specific simulation for autonomous surgery,” Ph.D. dissertation, UNIVERSITÀ DEGLI STUDI DI VERONA, 2022.

[392] A. Xie, F. Ebert, S. Levine, and C. Finn, “Improvisation through physical understanding: Using novel objects as tools with visual foresight,” arXiv preprint arXiv:1904.05538, 2019.

[393] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, “Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection,” The International Journal of Robotics Research, vol. 37, no. 4-5, pp. 421–436, 2018.

[394] A. Zhang, N. Ballas, and J. Pineau, “A dissection of overfitting and generalization in continuous reinforcement learning,” arXiv preprint arXiv:1806.07937, 2018.

[395] S. Ross, G. Gordon, and D. Bagnell, “A reduction of imitation learning and structured prediction to no-regret online learning,” in Proceedings of the fourteenth international conference on artificial intelligence and statistics, 2011, pp. 627–635.

[396] C. Yang, K. Yuan, Q. Zhu, W. Yu, and Z. Li, “Multi-expert learning of adaptive legged locomotion,” Science Robotics, vol. 5, no. 49, 2020.

[397] C. Devin, A. Gupta, T. Darrell, P. Abbeel, and S. Levine, “Learning modular neural network policies for multi-task and multi-robot transfer,” in 2017 IEEE international conference on robotics and automation (ICRA). IEEE, 2017, pp. 2169–2176.

[398] A. Pore and G. Aragon-Camarasa, “On simple reactive neural networks for behaviour-based reinforcement learning,” in 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 7477–7483.

[399] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, P. Abbeel, and W. Zaremba, “Hindsight experience replay,” arXiv preprint arXiv:1707.01495, 2017.

[400] M. Plappert, M. Andrychowicz, A. Ray, B. McGrew, B. Baker, G. Powell, J. Schneider, J. Tobin, M. Chociej, P. Welinder, V. Kumar, and W. Zaremba, “Multi-goal reinforcement learning: Challenging robotics environments and request for research,” arXiv:1802.09464v2[cs.LG], 2018.

Appendix A

Appendix: Hierarchical Task Decomposition using DRL for pick and place task

Robot learning has gained increasing attention in recent years, particularly with the development of DRL methods, which have demonstrated breakthroughs in dexterous manipulation [393], grasping [43], and navigation for locomotion tasks [42]. However, a significant challenge that hinders the universal adoption of DRL in robotics is the data-hungry training regime, which requires millions of trial and error attempts to learn goal-directed behaviors, making it impractical in real robotic hardware [394]. Furthermore, existing DRL methods learn complex tasks end-to-end, leading to overfitting of training idiosyncrasies and making them less adaptable to other tasks, which results in poor sample efficiency [394]. As a result, when solving problems that are highly similar to a pretrained task, new DRL policies have to be trained from scratch, which leads to a wastage of computation power.

Compared to end-to-end DRL methods, LfD approaches have been developed to be more efficient. These approaches involve training a neural network to replicate the expert's behavior from a dataset of reference trajectories. However, to achieve adequate training, a substantial number of demonstrations and specialized data-acquisition hardware and instrumentation, such as virtual reality or teleoperation units, are required [53]. LfD's efficacy is limited since it can only perform as well as the reference trajectory, without any additional feedback for improvement. Moreover, common LfD techniques such as BC are susceptible to compounding errors in long time horizon tasks [395].

An alternative approach to learning long time horizon tasks is through the use of HRL. HRL is a RL setting that enables training of multiple agents at varying levels of temporal abstraction [148]. This approach involves training low-level agents to encode primitive motor skills, while the higher-level policy selects which low-level agents are to be used to complete a task, following an end-to-end training paradigm [149, 150]. Beyret *et al.*. [151] proposed an explainable HRL method for a robotic manipulation task that employs HER as a high-level agent to decide on goals that are given as input to the low-level policy. Although hierarchical policies are learned end-to-end in these works, they often observe instability, leading to sample inefficiency, wherein the lower level policy changes under a non-stationary high-level policy.

In order to address the issue of unstable policy update in hierarchical policies, researchers have explored multi-subtask approaches that use modularization of neural networks to encode certain attributes of a complex control problem [396, 397]. These attributes are trained separately and combined in various ways to produce versatile behaviors. For instance, Yang *et al.* proposed a method that employs pre-trained motor skills parameterized by a DNN and fused them to generate various locomotion behaviors [396]. Similarly, Devine *et al.*. studied modular neural network policies for learning transferable skills across multiple tasks and robots [397]. Xu *et al.*. used parallel attribute networks to combine parallel skills simultaneously [145], while Pore *et al.*. trained individual subtask networks using BC and then combined them using a high-level DRL network [398]. One of the benefits of using subtask networks is that they are easier and faster to train compared to learning an overall control policy [145]. Additionally, modular behaviors are easier to interpret and can be adapted to similar tasks. However, designing subtask networks requires a priori knowledge of the task, which can be less demanding than expert demonstrations in LfD [396].

Therefore, we hypothesize that a complex control task can be simplified into high-level subtasks using the human operator's domain knowledge. These subtasks can then be learned using DRL techniques, allowing for a learned policy that considers the robot's environment and mechanical constraints rather than human bias from demonstrations. Specifically, we focus on a pick-and-place task and manually decompose it into three subtasks: approaching the object, grasping the object, and retracting the object to a target position. To train these subtasks, we use a low-level DRL policy called the Low-level Subtask Expert (LSE) that learns each subtask independently with a sub-goal directed reward function. To coordinate the subtasks, we employ a High-Level Choreographer (HLC) DRL policy that learns to sequence the subtasks to achieve the desired behaviors. Our proposed approach is illustrated in Fig. A.1. Previous research has also explored the use of modular subtask networks for complex control tasks [396, 397, 145, 398].

Our research contribution entails the development of a multi-subtask DRL approach for pick and place tasks, which we compare to an established LfD baseline. Moreover, we demonstrate the efficacy of our method by transferring the learned policies to a physical robotic system and evaluating its performance in grasping objects with various geometric shapes.

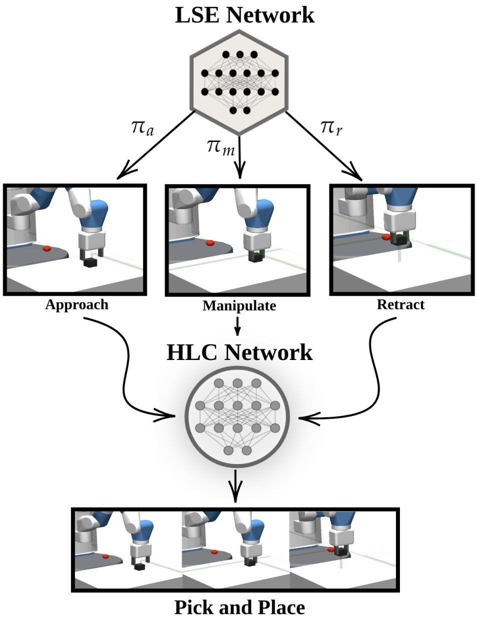


Fig. A.1 Summary diagram of the hierarchical architecture proposed in this paper. The pick and place task is divided into Low-level Subtask Experts (LSE), namely *approach*, *manipulate* and *retract*. These subtasks are coordinated using a High Level Choreographer (HLC).

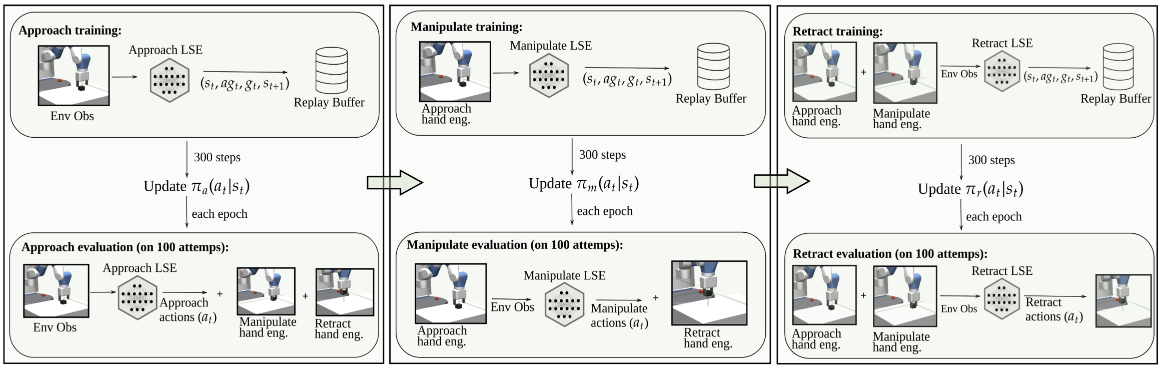


Fig. A.2 Schematic overview of the LSE training and evaluation process: All the LSE are trained independently (from left to right) approach, manipulate and retract respectively. The LSE policy π is updated offline by sampling from a replay buffer after 300 steps using DDPG+HER. The policy is evaluated after each epoch by using hand-engineered solution for other subtasks by computing the success rate on 100 episodes.

Training the Low-level Subtask Expert (LSE)

The goal of the LSE is to learn an optimal policy and task representations to perform specific subtasks. To achieve this, we formulate a MDP for LSE. Within each subtask u_i (where $i, 1 \leq i \leq 3$), at each time step t, the agent receives a state input S_t from the environment E, executes an action a_t, and transitions to the next state S_{t+1}. We use the Deep DDPG algorithm (Sec. 3.1.3) coupled with HER to train the LSE policy π_{u_i}, as it has been shown to be a promising approach for end-to-end pick tasks [399, 400].

The state inputs to the agent are vector observations that provide kinematic information, such as position, velocity, and orientation, of the object and the robotic gripper. The action output of LSE consists of x, y, and z positions. Each LSE is parameterized by a neural network that includes three fully connected layers and one final linear output layer. The network architecture can be found in the project code [1].

During the training process, for each subtask u_i, a list of tuples $(s_t, ag_t, sg_t, s_{t+1})$ is stored in the replay buffer at each episode (i.e., 300 steps), where s_t is the observation at the beginning of the episode, ag_t is the achieved goal after taking actions during the episode (i.e., the new gripper position), sg_t is the goal of the subtask during the episode, and s_{t+1} is the new state after completing the action in the environment. We design a dense reward function r_t that is defined as:

$$r_t = -d(ag_t - sg_t)$$

This function returns the negative Cartesian distance d between the achieved goal and the subtask goal at each timestep.

The DDPG+HER algorithm samples state observations from the replay buffer and updates π_{u_i} every 300 steps. After each epoch (15k steps), the performance of the LSE is evaluated using hand-engineered actions for the subtasks that are not being trained. Figure A.2 provides a schematic overview of the described method. Hand-engineered solutions are pre-configured action values used to reach a desired target state. For the evaluation process of the *approach* subtask, the action output from the LSE network is used, and hand-engineered actions are used for the *manipulate* and *retract* subtasks.

Thus, if at the end of the episode, the block fails to be placed at the target position, it implies that the *approach* part has not been successful and needs further training. Note that the engineered solutions are only used to reach an intended position before training a specific LSE module and for the evaluation phase to test whether the robot can successfully complete the task.

High Level Choreographer (HLC)

Once the LSEs are trained, a HLC is established to learn a policy that choreographs the subtasks to complete the task temporally. At a given timestep t', the HLC operates in state $s_{t'}$, selects a subtask u_i, and receives a reward $r_{t'}$ upon completion of the subtask. The agent then transitions to a state $s_{t'+1}$, which corresponds to the state after executing the subtask. Here, we use an actor-critic network architecture, as introduced in Sec. 3.1.3, where the actor

[1] Project code: https://github.com/LM095/DRL-for-Pick-and-Place-Task-subtasks

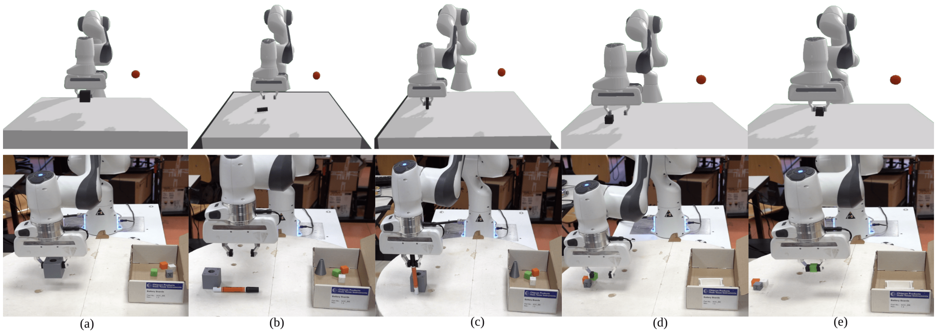


Fig. A.3 Pick and place task (a) accomplished with end-to-end learning strategy with DDPG+HER and our LSE DDPG+HER. (b) failure with a thin cylindrical object for end-to-end strategy (c) success with a narrow cylindrical object for the agent trained with our LSE strategy. (d) failure with a small box object for the agent trained with end-to-end strategy. (e) success with a small box object for the agent trained with our LSE strategy.

policy selects one of the subtasks [398]. The network consists of a recurrent layer followed by two independent, fully connected layers serving as the actor and critic.

As the output of the HLC is a discrete action value, we employ an asynchronous Actor-Critic (Advantage Actor-Critic (A3C)) training strategy to learn the HLC policy [118]. We define a *sparse* reward function $r_{t'}$, where the HLC receives a positive reward if the robot successfully places the block at the target position by selecting the correct subtask sequence.

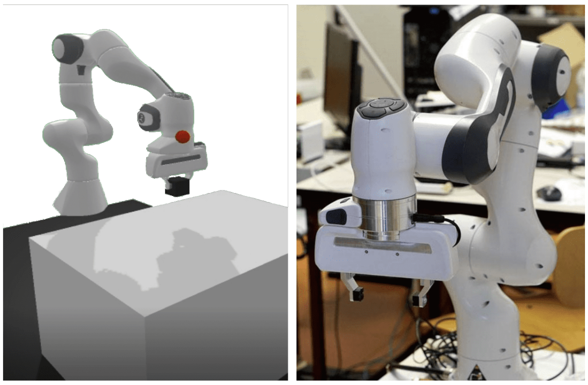


Fig. A.4 Different environments used for experiments (a) *PandaPickAndPlace-v0* (b) Franka Emika Robot used for real robot demonstrations.

Experiments

To evaluate the effectiveness of our proposed approach, we conducted two sets of experiments. Firstly, we conduct a comparative study between our proposed LSE approach and a baseline LSE trained via BC [398]. BC has been shown to be an efficient baseline compared to end-to-end DRL methods. Secondly, we demonstrate the successful translation of the learned policy from simulation to a real robotic system. The training methods are performed on an Intel Core i7 9th Gen system.

Simulation Experiments: PandaPickAndPlace-v0 We use the Mujoco simulation engine environment, *PandaPickAndPlace-v0*, which includes the Panda robot, as shown in Fig.A.4a. Once an LSE reaches a high success rate, the network weights are saved, and a similar strategy is used to train the remaining subtasks. After training all the subtasks, we load the network weights and train the HLC to choreograph the subtasks temporally. We compare the training performance of each subtask using two methods trained via DDPG+HER and BC, following the schematics shown in Fig.A.2.

Real Robot Experiments In the second part of our experiments, we establish communication between the simulation environment and the real robot using a ROS node, which interfaces with the *Moveit* framework [**?**]. The poses generated by the actions in the *PandaPickAndPlace-v0* environment are processed by *Moveit* to generate the complete trajectory while observing the physical constraints of the real robot. Furthermore, a homogeneous transformation is applied to change the reference frame, which lies at the gripper center in the simulation scene, to the panda base frame in the real robot.

Lastly, we demonstrate the reusability of the subtasks by fine-tuning the LSE to grasp different types of objects, such as a cylinder and a block of different dimensions, used in the training procedure (see Fig. A.3). An end-to-end learning approach would require complete retraining for different objects. The proposed LSE approach provides a possibility to change one of the subtasks without affecting other trained subtasks. Using a subset of behaviors is not possible in end-to-end learning. Therefore, in our proposed method, we use the trained LSE on the block pick-and-place task and fine-tune the grasping for the *retract* subtask, whereas we directly deploy the behaviors learned in the end-to-end learning.

Results

This study aimed to compare the performance of a DRL technique for training a LfD system, called LSE, with a supervised BC baseline. Fig. A.5 depicts the sample efficiency of the LSE strategy trained via DDPG+HER and BC learning methods. The peak represents the maximum success reached by each method for each subtask, where the first peak denotes the completion of training the *approach* subtask, the second peak denotes the completion of the training of *manipulate* subtask, and the third peak indicates the training of the *retract* subtask.

The results demonstrate that DDPG+HER outperforms BC, reaching 100% success in 218k steps, while BC takes 372k steps. Moreover, DDPG+HER shows a smooth, monotonous learning curve compared to BC, which does not stabilize immediately after reaching high success values. Overall, DDPG+HER shows less variance compared to BC. There is a significant difference between the learning curve for the *retract* behavior, which is a temporally

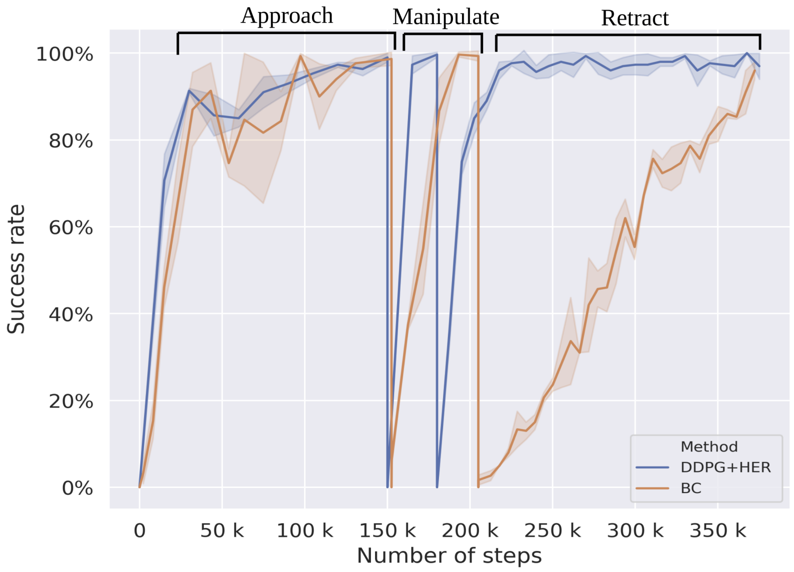


Fig. A.5 Performance comparison of our training strategy using DDPG+HER and BC. Each experiment is executed independently three times with different seeds. Success is quantified as the percentage of successful grasp as a function of training steps.

elongated subtask compared to other subtasks. Due to the long horizon task, BC seems to suffer from the compounding error caused by a covariate shift. Hence, we observe that DDPG+HER is faster in learning for the *retract* subtask.

Table A.1 shows the comparison of the training performance of the methods presented in this work. In particular, we analyze two possible strategies: a subtask approach using BC and a new methodology proposed in this paper. For the strategies that use subtasks, we define LSE1 as the *approach*, LSE2 as the *manipulate*, and LSE3 as the *retract*.

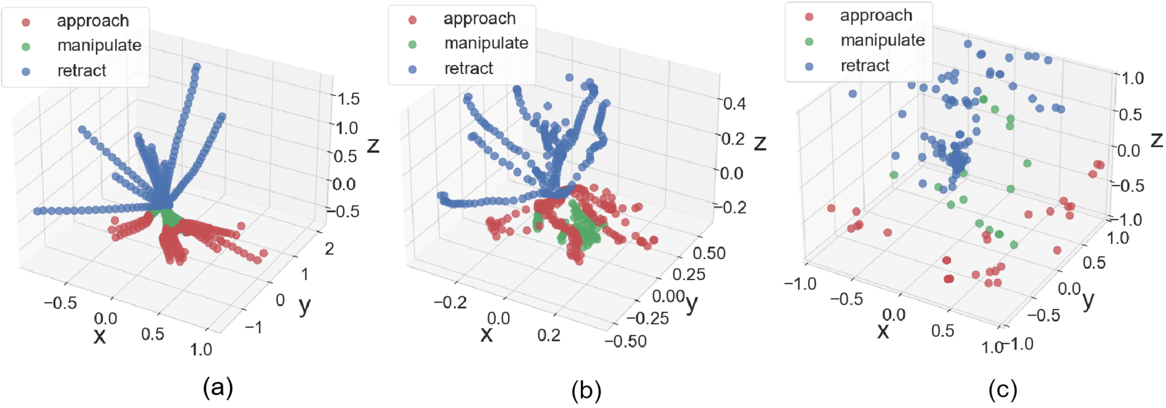
DDPG+HER using subtask decomposition is the best performing approach, and the results suggest that following the subtask approach, training can be more effective if we use a DRL algorithm than supervised BC. The behavior learned by DDPG+HER is more robust and does not require the collection of expert demonstrations, which can be time-consuming and often reflects less variability. Moreover, training using a subtask approach shows a significant reduction in both steps (by $\sim 77\%$) and time (by $\sim 75\%$) with respect to end-to-end training and therefore is the best training strategy in this context.

We compare the actions learned by a subtask-based LSE policy and an end-to-end policy. To this end, we analyze their activation patterns in the Cartesian space for ten episodes, using trained LSEs and the end-to-end model, respectively. The analysis is based on the premise that the initial environment conditions are the same for both policies.

As shown in Fig. A.6, the actions generated by the LSE networks are in the proximity of the hand-engineered actions, indicating that the learned behavior is specialized to the particular subtask. However, there is a slight deviation in the manipulate activation of

Table A.1 Performance of methods for the same level of success rate

	Number of steps					Total time
	LSE1	LSE2	LSE3	HLC	Total	
DDPG+HER end-to-end	-	-	-	-	1.4M	~1h
BC LSE	152k	52k	168k	98k	470k	~25 min
DDPG+HER LSE	150k	30k	**38k**	98k	**316k**	~18 min



(a) (b) (c)

Fig. A.6 LSE specialization analysis using different training strategies. Samples representing activation patterns using (a) hand-engineered solutions (b) learned using our subtask approach (c) learned using an end-to-end strategy for ten episodes.

hand-engineered and learned behaviors, which may be attributed to near-zero manipulation activations and overfitting. On the other hand, the network activations for the end-to-end approach do not exhibit any specific pattern, verifying our hypothesis that the LSE approach makes the task tractable compared to an end-to-end approach.

Furthermore, we conducted real robot experiments using the subtask approach and end-to-end training methods. The results, shown in Fig. A.3, indicate that using the subtask approach, the robot can pick up various objects, whereas using an end-to-end training method, the robot can only complete the block pickup for which it was trained and fails in grasping all other objects. Additionally, the LSE approach allows for fine-tuning of the gripper closure for a particular subtask, enabling the robot to grasp different types of objects that are not possible with an end-to-end policy. These findings confirm that the subtask approach can generate robust behavior by fine-tuning a subset of the subtask.

Publications linked to this chapter

1. Luca Marzari, Ameya Pore, Diego Dall'Alba, Gerardo Aragon-Camarasa, Alessandro Farinelli, and Paolo Fiorini.“Towards Hierarchical Task Decomposition using Deep Reinforcement Learning for Pick and Place Subtasks." In 2021 20th International Conference on Advanced Robotics (ICAR), pp. 640-645. IEEE, 2021.