

東京大学大学院工学系研究科機械工学専攻
令和 6 年度修士論文

眼科硝子体手術支援ロボットの自動化に
関する研究

Research on Autonomous Control of Vitreoretinal
Surgical Robot

指導教員

原田 香奈子 教授

学籍番号 37-225002

Yuen Christian Tomu Gee Seun

Research on Autonomous Control of Vitreoretinal Surgical Robot (眼科硝子体手術支援ロボットの自動化に関する研究)

発表者: Christian YUEN, 指導教員: Kanako HARADA

Vitreoretinal surgery, Internal limiting membrane peeling, Surgical robots, Keypoint detection, Automation

1. Research Background and Objective

Vitreoretinal surgery involves delicate procedures performed within the confined workspace of the eye, demanding exceptional precision to minimize risks such as retinal damage. Challenges such as physiological hand tremors and the inability to sense minimal forces highlight the limitations of manual surgery [1, 2]. These constraints drive development of robotic systems to enhance safety and accuracy.

The Vitreoretinal SmartArm system previously demonstrated its effectiveness in automated needle-based tasks using a shadow-based positioning framework [3]. However, adapting the system for forceps, essential for tasks such as internal limiting membrane (ILM) peeling, introduces new challenges. The forceps' dynamic geometry, additional degrees of freedom, and complex shadow behavior complicate detection and positioning.

This research focuses on integrating forceps into the Vitreoretinal SmartArm system as illustrated in Fig. 1, improving keypoint detection, and adapting the autonomous positioning framework for forceps. These advancements aim to address current limitations and lay the foundation for future robotic-assisted vitreoretinal surgeries.

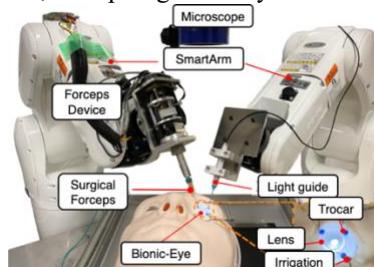


Fig.1 Vitreoretinal System

2. Advancing the Robotic Capabilities with Forceps Integration

2.1 Integration of the forceps system

The integration of the forceps introduced two additional degrees of freedom: rotation and grasping. Hardware integration includes an ILM-specialized forceps surgical instrument as shown in Fig. 2, a forceps device with motorized gear sets, a control box for motor control, and I/O boards for communication with the SmartArm system. These modifications enabled precise control of the forceps, tailored to meet the demands of ILM peeling.

2.2 Progressing the robot kinematic design
The kinematic framework was updated to accommodate the additional rotational degree of freedom, using a constrained optimization approach



Fig.2 The Forceps Device

[4]. The system was first validated in a simulation environment and later incorporated into the complex vitreoretinal surgical scene, and lastly with the real robots where the forceps was teleoperated within the Bionic-EyE ILM model,



Fig.3 ILM Grasping

as shown in Fig. 3.

2.3 System framework and codebase enhancements

The software framework underwent substantial restructuring, including the transitioning to the SmartArmStack for modern structuring and streamline workflow within the laboratory. Additional enhancements including code refactoring, dynamic variable for joints, and helper functions, creating a modern modular architecture that improves efficiency and future development.

3. Shadow-Based Positioning Framework

3.1 Keypoint Detection Neural Networks (NN)

3.1.1 Splitting Keypoint Detection Tasks

To address challenges arising from the forceps' complex geometry and dynamic orientation, a dual-neural network approach was implemented. The previous method, which relied on determining the ROI center point from the prior frame, often failed during misdetections, leading to the loss of the desired region. By separating keypoint detection and ROI management and relocating shaft keypoint detection to the ROI NN—capable of observing the entire microscope workspace—detection accuracy was significantly improved. Both networks were trained on 1,019 labeled images, augmented to 6,000, using a ResNet-18 encoder pretrained on ImageNet.

3.1.2 Redesigning Keypoint Detection for Forceps

A dedicated keypoint detection NN was developed to accurately identify the forceps tip and shadow tip, accommodating the instrument's unique geometry. Enhanced pre- and post-processing techniques were integrated to ensure reliable detection under varied environmental and operational conditions.

3.1.3 Adaptive ROI and Shaft Point Detection

This network identifies the ROI center point and shaft keypoint, allowing the system to recover the detection region even when the forceps move outside the ROI. By enabling keypoint detection from any position within the workspace, the system ensures consistent performance across diverse scenarios.

3.2 Autonomous Positioning and Image Processing

The autonomous positioning program was extended to include the orbital manipulation kinematics framework [5], with compatibility between both simulation and real robots. The image processing node, depicted in Fig. 4, was restructured to improve runtime efficiency and facilitate seamless integration of the neural networks.

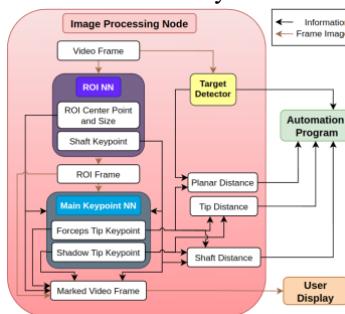


Fig.4 Image Processing

The node provides planar distance, tip distance, and shaft distance to the automated program. Code refactoring further enhanced performance and maintainability for future developments.

4. Shadow-Based Autonomous Positioning with Forceps Experiment

4.1 Experimental Setup

The experiment evaluated the shadow-based autonomous positioning framework with the forceps, using the established four-step positioning strategy: planar positioning, overlap prevention, vertical positioning, and additional positioning. A trial was deemed successful if the system attempted or successfully contacted all five target points without requiring operator intervention.

4.2 Results

The system achieved a success rate of 33% (4 out of 12 trials) and successfully contacted

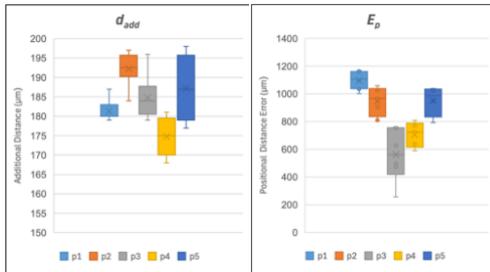


Fig.5 d_{add} and E_p

29.4% (10 out of 34) of the target points, underscoring challenges in vertical positioning (additional distance, d_{add}) and lateral alignment (positioning error, E_p), especially at workspace boundaries as depicted by Fig. 5. Key challenges included difficulties in keypoint detection due to forceps shadow occlusion, underrepresentation of specific light and forceps orientations in the trained dataset, and

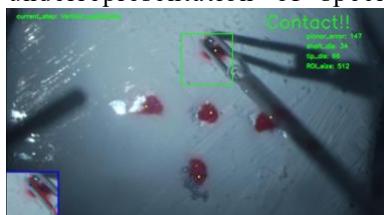


Fig.6 Excessive force

environmental factors. Unsuccessful trials were often caused by errors during the vertical positioning step, requiring operator intervention as shown in Fig. 6.

4.2 Discussion

The results provided valuable insights into the system's performance, underscoring the need for further improvements in detection to enhance overall reliability. Comparing the original needle-based

experiment with the forceps (Table 1), success rates, and errors show there is much room for improvement.

Table 1 Comparison of Positioning Results

Instr.	# Trials	$d_{add}(\mu\text{m})$	$E_p(\mu\text{m})$	Dur.(s)
Needle	5	310	600	257
Forceps	4	183	880	385

5. Proposed Experiments and Methods

5.1 Preliminary Exploration for Forceps Alignment

To further address forceps-specific challenges, this research explored a preliminary method for aligning the forceps rotationally to minimize force application during ILM peeling. By ensuring that both tips of the forceps make simultaneous contact with the retina, the proposed alignment method reduces the risk of excessive force and potential retinal damage.

5.2 Experimental Plan for ILM Peeling

A structured experimental plan for evaluating ILM peeling was also outlined, utilizing Bionic-Eye models and combining surgeon training, performance metrics, and system validation.

6. Conclusion and Future Works

By advancing the capabilities of the Vitreoretinal SmartArm system, this research demonstrates the feasibility of robotic-assisted ILM peeling while addressing key limitations of the existing framework. The integration of forceps, enhancements to the positioning framework, and proposed alignment methods highlight critical steps toward improving safety, precision, and efficiency in robotic vitreoretinal surgery.

Future work aims to advance the hardware and software for more accurate, robust, and faster image processing to implement (semi-)autonomous methods.

References

- [1] Gupta, P. K. et al., "Surgical forces and tactile perception during retinal microsurgery," *Med. Image Comput. Comput.-Assist. Intervent.*, Springer, pp. 1218–1225.
- [2] Singh, S. et al., "Physiological tremor amplitude during retinal microsurgery," *Proc. IEEE Northeast Bioeng. Conf.*, pp. 171–172.
- [3] Koyama, Y. et al. "Autonomous coordinated control of the light guide for positioning in vitreoretinal surgery," *IEEE Trans. Med. Robot. Bionics*, vol. 4, no.1, pp.156-171, Jan. 2022.
- [4] Marinho, M. et al., "A unified framework for the teleoperation of surgical robots in constrained workspaces," *2019 Int. Conf. Robot. Autom. (ICRA)*, pp. 2721–2727, ISSN: 2577-087X.
- [5] Koyama, Y. et al. "Vitreoretinal Surgical Robotic System with Autonomous Orbital Manipulation using Vector-Field Inequalities," *2023 Int. Conf. Robot. Autom. (ICRA)*.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to all who have been part of my journey in Japan and have supported me throughout this research. I never would have thought I would meet this many amazing people from around the world as it has been a pleasure. It has been an amazing experience and great honor to have been part of the most prestigious university in Japan. My experiences here will always be a part of me as I consider Japan my home away from home.

I would like to give Professor Kanako Harada, my supervisor, my upmost appreciation as she gave me this opportunity to research my interest of robotics in her laboratory. She has always been supportive and kind to everyone in her lab, making it a pleasant environment to be a part of. She has also provided concise and helpful guidance, enabling me to efficiently produce great work. I am always impressed by her hard work, dedication, and skill to receive funding for her lab. I would not have had this opportunity without her.

I would also like to give Professor Murilo Marques Marinho my thanks, as he has been a great source of knowledge and help. Even though he departed from our lab, he still often visits with great intentions, continuing collaborations between our labs. He is very admirable and knowledgeable in his work as if it were like the back of his hand. My research would not have been possible without all of the foundational research he has provided.

I would like to also give my appreciation to Assistant Professor Saul Heredia, as he has

given me great feedback and support. His knowledge and expertise in VR simulation has greatly contributed to our labs research, especially for future development of automated systems. He is definitely an indispensable figure in the lab as he puts in the hard work and is always willing to help the students.

To all the participants who have taken part of the monthly eye surgery meetings, I would like to give my thanks for spending the time and attention to provide valuable feedback in my work. Doctor Ueta has always provided very insightful information and feedback that has driven the direction of my research. Professor Mitsuishi has also provided great feedback and was a crucial member of establishing our lab.

I would like to thank the Japanese Government for providing me early financial support at my time at the university, providing a peace of mind staying financially sound.

To all of the professors and sensei's who have taught the lectures and classes I have been a part of, thank you for sharing your knowledge. Even though my main focus at the university was for my major of mechanical engineering, Japanese classes were always fun to be a part of.

To all the current/past student and post-doctoral members of the lab, I appreciate your support and companionship throughout the years. Juan-san helped me understand robot kinematics and I hope he has been doing well upon the return to his home country. Koyama-san provided me his Vitreoretinal SmartArm platform with guidance, enabling me to conduct my research. Xiaofeng-san has been an overall great help and has always worked by my side (literally). PhD students, including Quentin-san, Zhao-san, Yifan-san, and Song-san have been always a great resource to learn and ask from. To all the other students, Lok-san, Kai-san, Kim-san, Yoon-san, Unosawa-san, Can-san, Tanaka-san, Li-san, Elia-san, Luis-san, and among others, I would like to thank for the great times and struggles we all experienced together.

The administration at the university has always been kind and efficient processing and

helping me with my documentation. I would like to thank Kozawa-san, Itagaki-san, Bai-san, Koike-san, and Ishii-san within our lab, as they have always helped me kindly.

To my family, I appreciate all of the different forms of support you have provided me, including emotional support, financial support, and encouragement.

To all of my friends I have made in Japan, I am grateful for all of the fun trips, events, banter, long nights, and drinks we have experienced together.

To my friends back home, I thank you for continuing to reach out and continue our friendship even though we are many freedom units away from each other.

Thank you again for everyone's collective support as you have all been a large part of this chapter in my life. I wish the best of luck to the current and future students with their research as it requires dedication and hard work. I also wish everyone the best of luck and a bright future as their support should be recognized. I hope that we will continue to share a future together in one way or another.

Christian Yuen, The University of Tokyo, January 2025

Contents

Acknowledgements	iii
Chapter 1 Introduction	1
1.1 Research background	1
1.1.1 Microsurgery	1
1.1.2 Vitreoretinal Surgery	2
1.1.3 Related Works	3
1.1.4 The Vitreoretinal SmartArm system	5
1.2 Research Objective	7
1.3 Contributions of this Research	8
1.3.1 Theoretical Contributions	8
1.3.2 Practical Contributions	8
1.3.3 Organization of this Work	9

Chapter 2	Background and System Overview	11
2.1	Robotic Systems	11
2.1.1	History of Surgical Robotic Systems	11
2.1.2	Current Surgical Robotic Systems	13
2.1.3	SmartArm Surgical Robotic System	19
2.2	Vitreoretinal SmartArm System	21
2.2.1	Hardware and Software Architecture of Previous Work	22
2.2.2	Shadow-Based Autonomous Positioning	25
2.2.3	Vitreoretinal Task Constraints	27
2.2.4	Image Processing the Instrument Tip and its Shadow	29
2.2.5	Comparing Different Levels of Autonomy	30
2.2.6	Orbital Manipulation	34
2.3	Limitations of the Original System	35
2.3.1	Challenges in Forceps Integration	36
2.3.2	Limitations in the Image Processing Pipeline	37
2.3.3	Maintainability and Workflow Inefficiencies	37
2.4	Internal Limiting Membrane Peeling Background	39
2.4.1	What is ILM Peeling?	39

2.4.2	ILM Peeling Procedure and Techniques	41
Chapter 3	Advancing the Robotic Capabilities with Forceps Integration	45
3.1	Integration of the Forceps System	45
3.1.1	The Forceps Primary Surgical Instrument	47
3.1.2	Forceps Robot Attachment	48
3.1.3	Control Box	50
3.1.4	Forceps Computer	52
3.1.5	Escon Controller	53
3.2	Progressing the Robot End Effector and Its Kinematic Design	54
3.2.1	The Seventh Joint	54
3.2.2	Adding Rotation to the Constrained Optimization Algorithm .	55
3.2.3	Simulator Integration	56
3.2.4	Teleoperating the Forceps	57
3.3	System Framework Enhancements and Code Optimization	59
3.3.1	Streamlining System Architecture and Workflow Optimization	59
3.3.2	Code Refactoring	60
Chapter 4	Shadow-Based Positioning Framework	63
4.1	Rationale for Splitting Keypoint Detection	64

4.2	Redesigning the Main Keypoint Detection for Forceps Adaption	65
4.2.1	Adapting the Neural Network	65
4.2.2	Adjusting the ROI Size	66
4.2.3	Pre-Image Processing for Enhanced Keypoint Detection	67
4.2.4	Dataset and Augmentation	68
4.2.5	NN Training Results	68
4.2.6	Enhancing Keypoint Detection Stability and Noise Mitigation .	70
4.3	Adaptive ROI and Shaft Point Keypoint Detection	71
4.3.1	Lightweight Methodology and Neural Network Design	72
4.3.2	Integration of Shaft Keypoint Detection	72
4.3.3	ROI NN Training Results	73
4.3.4	Enhancements for Stability and Robustness	73
4.4	Advancements to the Autonomous Positioning Program	75
4.4.1	Autonomous Positioning Process Improvements	75
4.4.2	Detecting the Target Points First	76
4.5	Integration of the Image Processing Node and Autonomous Positioning	76
4.5.1	Image Capturing Node Enhancements	77
4.5.2	Asynchronous Processing	79

4.5.3	Architecture for Shadow-Based Autonomous Positioning	79
4.6	Shadow-Based Autonomous Positioning with Forceps Experimental Setup	81
4.6.1	Objectives	81
4.6.2	Methodology	81
4.6.3	Metrics for Evaluation	83
4.6.4	Control and Comparison	84
Chapter 5	Experimental Results and Discussion	87
5.1	Results	87
5.1.1	Quantitative Results	87
5.1.2	Qualitative Observations	89
5.2	Discussion	95
5.2.1	Observations Analysis	95
5.2.2	Comparison to the Original Experiment	96
5.2.3	Challenges and Limitations	98
5.2.4	Implications and Future Direction	98
Chapter 6	Proposed Experiments and Methods	101
6.1	Preliminary Exploration for Forceps Alignment	101

6.1.1	Importance of the Alignment Process	101
6.1.2	Future Development and Evaluation	102
6.2	Experimental Plan for ILM Peeling	103
6.2.1	Experiment Overview	103
6.2.2	Anticipated Challenges	105
6.2.3	Expected Outcomes	107
6.3	Discussion	107
Chapter 7	Conclusion and Future Work	109
7.1	Conclusions	109
7.1.1	Integration of the Forceps	109
7.1.2	Shadow-Based Autonomous Positioning with Forceps	110
7.1.3	Experimental Results and Discussion	111
7.1.4	Proposed Experiments and Methods	111
7.2	Future Work	112
Chapter 8	Appendix	115
8.1	Dual quaternions and operators	116
8.1.1	The dual number set, \mathbb{D}	116
8.1.2	The quaternion set, \mathbb{H}	117

8.1.3	The pure quaternion set, \mathbb{H}_p , and vectors in \mathbb{R}^3	119
8.1.4	The unit quaternion set, \mathbb{S}^3 , and rotations	120
8.1.5	Mapping quaternions and \mathbb{R}^4	122
8.1.6	The dual quaternion set, \mathcal{H}	123
8.1.7	The pure dual quaternion set, \mathcal{H}_p	124
8.1.8	The unit dual quaternion set, \mathcal{S} , and rigid body motion	125
8.1.9	Mapping dual quaternions and \mathbb{R}^8	126
8.2	Robot kinematics using unit dual quaternions	126
8.2.1	Forward kinematics model	127
8.2.2	Forward kinematics models for pose, translation, and rotation	127
8.2.3	Denavit-Hartenberg parameters	129
8.2.4	DENSO Corporations' proprietary parameters	130
8.2.5	Inverse kinematics	130
8.2.6	Differential Kinematics	131
8.2.7	The unit dual quaternion Jacobian, J_x	133
8.3	Constrained optimization algorithm	134
8.4	Vector-field-inequalities method	136
	Bibliography	139

List of Figures

1.1	The different components involved in vitreoretinal surgery: dominant instrument, light guide, trocar, and irrigation.	2
1.2	Internal limiting membrane (ILM) peeling with forceps instrument.	3
1.3	The Vitreoretinal SmartArm system, featuring software-based RCM, shadow-based positioning, and autonomous lighting for enhanced precision, safety, and functionality [12].	6
2.1	Roadmap of the advancement of surgery with robot integration over time [14].	13
2.2	The da Vinci® 5 video cart (left), the da Vinci® 5 surgeon console (middle), and the da Vinci® 5 patient side cart (right). (Images® 2024 Intuitive Surgical, Inc).	14
2.3	The hinotori™ surgeon cockpit (a), the operation robot unit (b), and monitor cart (c) (Images © 2022 Medicaroid Corporation).	15
2.4	The Senhance® Surgical Robotic System (Images © 2024 Asensus Surgical USA, Inc).	16
2.5	The PRECEYES Surgical System R1.1 (Images © 2024 PRECEYES B.V.). .	18

2.6	The Luca™ Surgical Robotic System being used during clinical trial (Images © 2024 AcuSurgical SAS).	18
2.7	The MUSA-3 robotic arm cart (left) and surgeon console (right) (Images © 2024 Microsure NL).	19
2.8	The 6-stage classification for assessing the autonomous capabilities of surgical robots [29].	21
2.9	The Hardware Architecture of the Vitreoretinal SmartArm system. A central control PC manages the operator and patient sides of the system [12].	23
2.10	The Software Architecture of the Vitreoretinal SmartArm system [12].	24
2.11	Visualized four-step shadow-based autonomous positioning strategy [26].	27
2.12	The geometrical primitives and types of VFI's used to generate each task constraint in the Vitreoretinal SmartArm system [26].	28
2.13	(a) The instrument shadow tip remains visible when the instrument's tip is within the light guide cone and microscope view. (b) The instrument shadow tip is properly illuminated when it remains within the illumination cone [26].	29
2.14	The marked image displayed to the user during the shadow-based autonomous positioning procedure [12].	31
2.15	Experimental setup using an Arduino system for contact detection and a QCR force sensor for force measurement [12].	32
2.16	Side view of positioning with and without orbital manipulation in simulation [27].	35

2.17	Microscope view of positioning without (a) and with (b) orbital manipulation in the real system [27].	36
2.18	Comparing the needle and forceps in the ROI view for keypoint detection.	38
2.19	Misdetection of the needle keypoints required for the ROI center point, resulting in the failure of reacquisition of the desired region.	39
2.20	OCT images of two patients who developed full-thickness macular hole postoperatively and closing months later [37].	40
2.21	The diagrams illustrate two methods for fovea-sparing internal limiting membrane (ILM) peeling, with and without an inverted flap [38].	42
3.1	The hardware setup of the Vitreoretinal SmartArm system with integrated forceps device.	46
3.2	The software setup of the Vitreoretinal SmartArm system with integrated forceps device.	47
3.3	The primary surgical instrument used in this thesis: Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga.	48
3.4	Dimension diagram of the the Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga.	49
3.5	The forceps robot attachment with its rotational and grasping capabilities.	49
3.6	The forceps control box containing the two motor positioning controllers.	51
3.7	Circuit diagram of the control box connected to the forceps attachment and I/O boards.	52

3.8	The Escon operation check node with motor control and state information.	54
3.9	Bug involving the implementation of rotation to the forceps with the Orbital Manipulation kinematics framework. The eyeball slightly shifts to a side and does not return to center.	56
3.10	A simple simulation scene of the robot manipulator with forceps device. The different axis of the end effector are shown.	57
3.11	Vitreoretinal workspace scene in Coppelia simulator with both tools inserted at the RCM points. The RCM points are set in this position before translating to the initial positions.	58
3.12	The forceps grasping the ILM inside the Bionic-Eye ILM model.	58
3.13	Updated End Effector Calibration UI with robot selection and calibration file update button	60
4.1	Different orientations the updated NN must recognize.	66
4.2	Preprocessed and augmented images for keypoint detection NN training.	69
4.3	Main Keypoint Detection Model Training and Validation Results - The model trained stably and does not drastically overfit.	69
4.4	Example of severe keypoint detection error of the forceps tip.	71
4.5	ROI NN Model Training and Validation Results - the model converged successfully and minimally overfits.	73
4.6	The ROI system is capable of self-correction, even when the forceps tip and its shadow are positioned near or slightly outside the bounding box, by employing a recovery function after detecting frozen frames.	74

4.7	Target point detection before instruments move to the starting position points.	77
4.8	Workflow of the Image Processing Node and Autonomous Positioning Program.	80
5.1	Box and whisker plots of the five target points in the shadow-based autonomous positioning with forceps experiment.	89
5.2	Notable keypoint detection issues.	91
5.3	ROI boundary issues affecting keypoint detection during p_1 positioning. The shadow is not within the ROI bounding box.	91
5.4	Minor inaccuracies for keypoint detection.	92
5.5	Excessive force exerted on the fundus leading to aborting the program during vertical positioning of p_1 .	92
5.6	Sudden instrument movements causing workspace shifts.	93
5.7	Different environmental factors affecting image processing.	93
5.8	Bubble on the cornea obscuring the image.	94
5.9	The rotational kinematics bug can be seen on the right side of the workspace view. The eyeball slightly rotates without returning to center.	99
6.1	Comparison of poor and optimal forceps alignment relative to the retina surface.	102
6.2	The Alcon® Grieshaber® Maxgrip Forceps 25 Ga+, a pointed, center-less forceps device.	104

6.3	ILM peeling task simulator developed by our group.	105
-----	--	-----

List of Tables

2.1	The median values of the five trials in each setup.	33
2.2	Results of autonomous positioning by FA.	34
3.1	Specifications of the the Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga.	48
3.2	Specifications for the rotation and grasping Maxon Group motorized gear sets.	50
5.1	Results of shadow-based autonomous positioning with forceps.	88
5.2	Comparison of the mean values of successful trials for the needle and forceps experiments.	97
8.1	Summary of distance functions and corresponding Jacobians.	138

Listings

Chapter 1 Introduction

1.1 Research background

1.1.1 Microsurgery

Microsurgery involves highly specialized procedures performed on small structures within the body such as blood vessels, nerves, and tissues using a microscope and precise instruments. Its minimally invasive nature offers significant benefits, including reduced trauma, scarring, and recovery time. This approach has found applications across various medical fields, including plastic, vascular, neurosurgery, orthopedics, and ophthalmology [1].

Advancements in technology have enabled supermicrosurgery, which targets even smaller structures, such as vessels and nerves between 0.3 to 0.8 mm in diameter [2]. However, as precision requirements increase, so do the challenges for surgeons, including eye strain, fatigue, and involuntary hand tremors. Physiological hand tremors, with an average amplitude of approximately 100 μm [3], can exceed the target size for microsurgical tasks, making manual precision difficult to achieve. Additionally, surgeons often struggle to sense forces as low as 7.5 mN [4], increasing the risk of unintended damage to delicate tissues.

These challenges highlight the limitations of human dexterity in tasks requiring micron-level precision, motivating the integration of robotic systems to assist in overcoming these

barriers.

1.1.2 Vitreoretinal Surgery

In ophthalmology, vitreoretinal surgery addresses disorders of the retina and vitreous, aiming to restore or preserve vision. These procedures are performed within a highly confined workspace observed through an ophthalmic microscope. Access to the interior of the eyeball is facilitated by trocars^{*1}, which allow the introduction of surgical instruments and irrigation systems^{*2}. The surgeon typically uses a dominant instrument, such as forceps or a microneedle, while a light guide illuminates the workspace. Figure 1.1 illustrates the key components involved in vitreoretinal surgery.

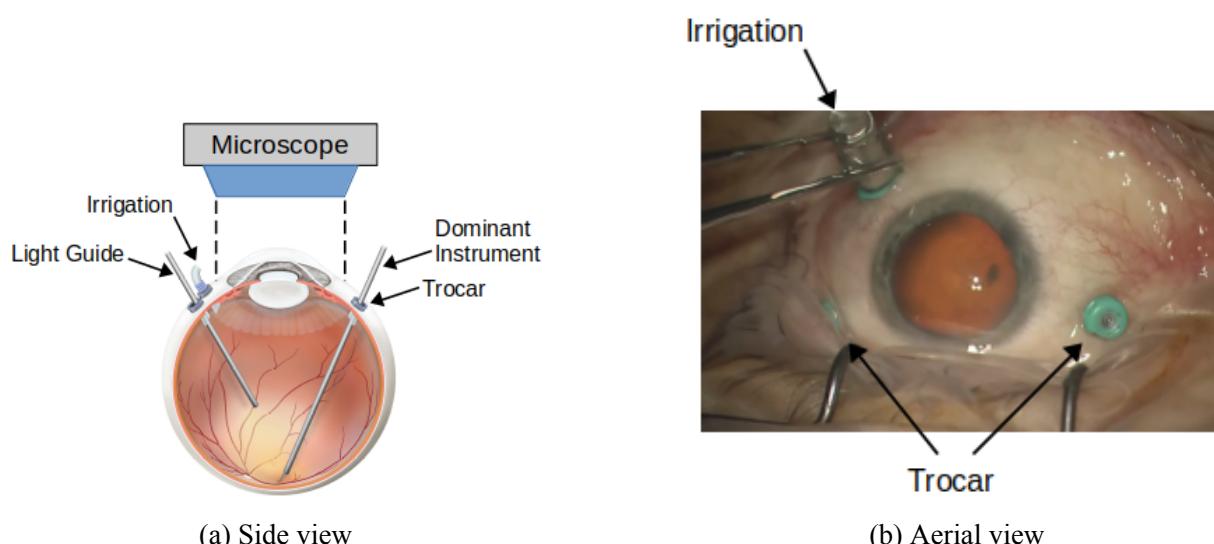


Fig. 1.1: The different components involved in vitreoretinal surgery: dominant instrument, light guide, trocar, and irrigation.

These procedures require exceptional precision due to the confined fluid-filled workspace, high magnification, the inherent difficulty of depth estimation, and the fragility of the tissues

^{*1}Typically 20 Ga to 25 Ga

^{*2}Balanced Salt Solution (BSS) is commonly used as the irrigation fluid.

involved. Internal limiting membrane (ILM) peeling exemplifies this challenge, as it involves the delicate removal of a thin, transparent retinal membrane. To prevent complications such as hemorrhage, the task demands meticulous control, making it one of the most complex procedures in vitreoretinal surgery. Further details on ILM peeling are provided in Section 2.4 and illustrated in Figure 1.2.



Fig. 1.2: Internal limiting membrane (ILM) peeling with forceps instrument.

1.1.3 Related Works

Robotic systems have significantly advanced vitreoretinal surgery by enhancing precision, reduce surgeon fatigue, reducing tremors, and enabling safer procedures. Early systems, such as leader-follower teleoperation platforms, addressed key challenges like manual dexterity and limited force sensing by stabilizing movements and providing haptic feedback to the operator.

The most notable effort, the PRECEYES Surgical System, is the first clinically validated vitreoretinal robot system that has demonstrated its viability and safety of clinical intra-ocular robotic-assisted surgery [5]. More information about the system is discussed in Section 2.1.2.

However, newer advancements are pushing the boundaries toward semi-autonomous and autonomous capabilities to further improve safety and performance.

- **Imaging-Driven Platforms:** Systems such as those developed by *Johns Hopkins University* integrate advanced imaging modalities like microscope-integrated OCT and RGB-D views to assist in subretinal injections. However, these systems are primarily optimized for needle-based procedures and lack adaptations for tools requiring more complex kinematics, such as forceps [6, 7].
- **Force-Control Systems:** The Ophthalmic Microsurgery Robot (OmSR) by *Jiaotong University and University College London* employs vision-force-based compliance control and a mechanical RCM. While effective for tasks like membrane peeling and needle insertion, it lacks vertical positioning capabilities for delicate operations requiring precise depth adjustments [8, 9].
- **Precision-Tracking Systems:** The *COLIBRI robotic platform* by the Technical University of Munich incorporates markerless 3D pose tracking to reduce RCM misalignment and forces on delicate tissues. However, this robot control method is optimized for navigation outside the eye [10].
- **Perception-Enhanced Platforms:** Researchers at the *Chinese Academy of Sciences* developed Multimodal Surgical Process Recognition (MSPR-DKS) to address perception challenges, including uneven lighting and motion blur. Their approach improves intraoperative precision during ILM peeling and builds a dataset for macular hole surgery [11].

These advancements illustrate the field’s trajectory toward safer, more precise, and intelligent robotic systems capable of addressing the inherent challenges of vitreoretinal surgery. History and further information about robotics in the surgical field are discussed in Section 2.1.

While these systems demonstrate significant advancements, several challenges remain, including:

- Limitations in accommodating surgical instruments requiring multiple DoF, such as forceps.
- Lack of precise depth control or vertical positioning in certain platforms, such as

OmSR.

- Developing intuitive control systems that reduce surgeon workload while maintaining precision.

1.1.4 The Vitreoretinal SmartArm system

Building on the advancements described earlier, the Vitreoretinal SmartArm system [12], as shown in Figure 1.3, addresses persistent challenges in robotic vitreoretinal surgery, such as precise depth estimation, workspace safety, and intuitive control. Unlike existing platforms, our system incorporates several novel features that enhance its adaptability, precision, and ease of use.

The Vitreoretinal SmartArm system is characterized by the following capabilities:

- **Software-Based RCM:** While platforms like PRECEYES, OmSR, and SHER rely on mechanical RCMs, our system implements a software-based RCM, offering greater flexibility and adaptability to dynamic surgical scenarios.
- **Vector Field Inequalities (VFI):** To enhance safety, the system employs VFIs to monitor and regulate interactions between surgical instruments and the workspace in real time.
- **Shadow-Based Depth Estimation:** A shadow-based positioning strategy was developed to precisely estimate the needle's depth within the workspace.
- **One-Hand-Off System with Autonomous Lighting:** By autonomously orienting the light guide relative to the surgical instrument, the system reduces surgeon workload, as only one hand is required for teleoperation.
- **Orbital Manipulation:** The system performs automated eye rotation by manipulating RCM points, improving access to difficult surgical sites and expanding the workspace view.

Further details and previous research with the original Vitreoretinal SmartArm system is

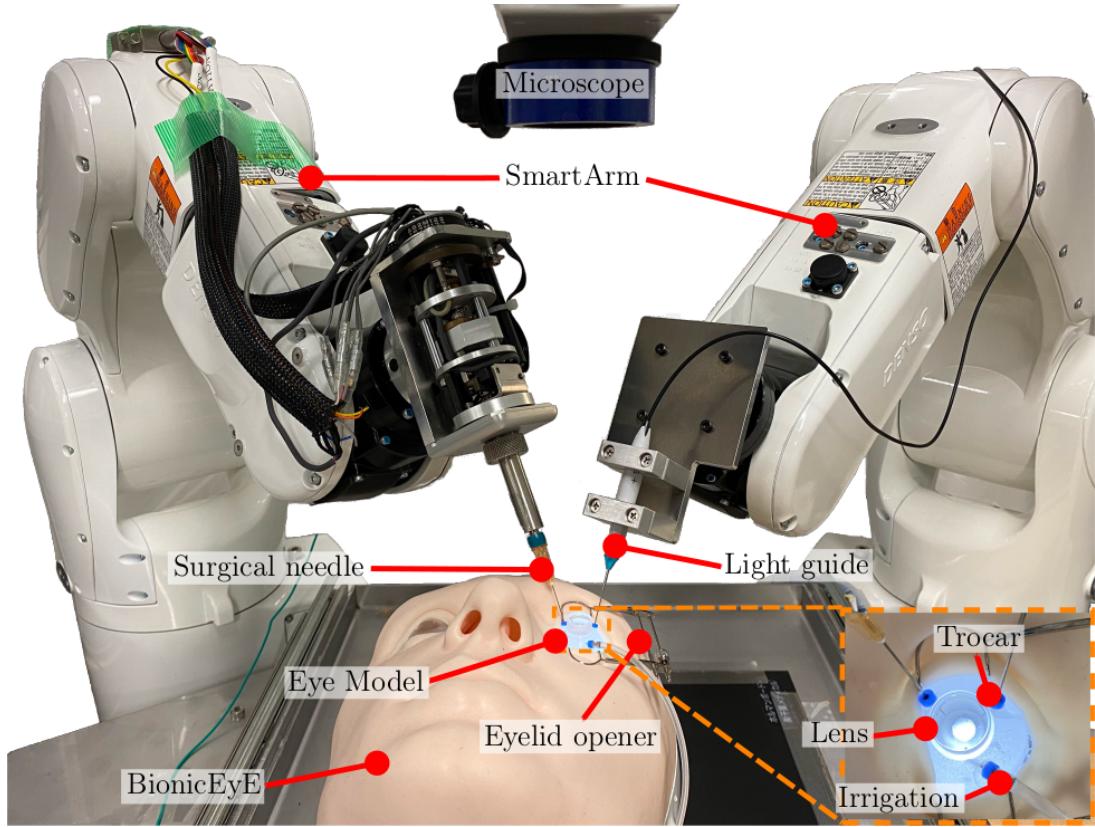


Fig. 1.3: The Vitreoretinal SmartArm system, featuring software-based RCM, shadow-based positioning, and autonomous lighting for enhanced precision, safety, and functionality [12].

discussed in Section 2.2.

While these features were highly effective for needle-based procedures, the system faced significant challenges when adapting to forceps, which require additional functionalities and considerations:

- **Increased Degrees of Freedom (DoF):** Forceps require opening, closing, and rotational movements, necessitating a redesign of the system's kinematics and control strategies.
- **Complex Shadow Geometry:** The forceps' dynamic grasping and geometry introduced complexities in the image processing framework, which was originally optimized for needle-based recognition.

These limitations highlighted the need for significant enhancements to the Vitreoretinal SmartArm system to accommodate forceps as the primary surgical instrument, forming the basis for the research objectives outlined in the next section. The limitations of the original system is discussed further in Section 2.3.

1.2 Research Objective

The original Vitreoretinal SmartArm system was optimized for needle-based procedures, offering precise depth estimation and workspace safety through its software-based RCM and shadow-based positioning framework. However, adapting the system for ILM peeling with forceps presented unique challenges, including the need for additional degrees of freedom and dynamic shadow behavior.

This research focuses on addressing these limitations by developing a fully integrated forceps system within the Vitreoretinal SmartArm platform. The specific objectives of this research are as follows:

- Design and integrate a forceps device into the robotic system, equipped with motors and controllers to support opening, closing, and rotational movements.
- Enhance the image processing and shadow-based positioning frameworks to accommodate the forceps' unique geometry and dynamic behavior.
- Evaluate the enhanced framework with the forceps and the improved image processing node for shadow-based positioning.
- Explore the development of an assistive method for precise and safe ILM grasping, aiming to minimize risks such as retinal hemorrhage and improve patient outcomes.
- Propose and outline an experimental plan using real robots and realistic phantom models to evaluate the feasibility of ILM peeling with the forceps device.

The findings from this research aim to advance the capabilities of robotic systems in vitreoretinal surgery, bridging the gap between manual techniques and fully automated solutions,

while laying the groundwork for future advancements in ILM peeling.

1.3 Contributions of this Research

1.3.1 Theoretical Contributions

This thesis makes the following theoretical contributions to the field of robotic vitreoretinal surgery:

1. **Exploration of Forceps Alignment Method:** Initiated the development of a forceps alignment method relative to the retinal surface to minimize unnecessary force during ILM peeling. This preliminary work underscores the importance of optimal rotational positioning for enhancing safety and precision.
2. **Experimental Framework for ILM Peeling Evaluation:** Proposed a structured experimental plan to assess the feasibility of the robotic system for ILM peeling tasks using realistic Bionic-Eye models. This plan serves as a foundation for future experimental validation of robotic-assisted ILM peeling.

1.3.2 Practical Contributions

This thesis makes the following practical contributions to the field of robotic vitreoretinal surgery:

1. **Development of a Forceps-Compatible Framework:** Designed and implemented a kinematic and keypoint detection framework specifically tailored for forceps integration. This framework adapts the existing shadow-based positioning system to address the complexities of the forceps' geometry, dynamic orientations, and functional requirements.
2. **Experimental Demonstration of Forceps Integration:** Conducted an experimen-

tal evaluation of the forceps-integrated system, showcasing the feasibility of shadow-based positioning with the forceps. The results provided valuable insights into the system's performance, limitations, and opportunities for enhancement.

3. **Advancing Robotic-Assisted ILM Peeling:** Laid a foundation for the further development of semi-autonomous and fully autonomous robotic methods on the Vitreoretinal SmartArm system. By addressing key challenges in integrating forceps, this work sets the stage for more advanced robotic solutions aimed at intricate surgical tasks like ILM peeling.

1.3.3 Organization of this Work

This thesis is organized into seven chapters, each addressing a critical aspect of the research. The structure provides a cohesive narrative, starting with an overview of the field and the robotic platform, progressing through the integration of the forceps system, and culminating in experimental evaluations and future directions. A brief overview of each chapter is provided below:

Chapter 2: Explores the field of surgical robotics, highlighting advancements and ongoing challenges. This chapter transitions into a detailed discussion of the Vitreoretinal SmartArm system, identifying its limitations and contextualizing the research within the challenges and clinical significance of the ILM peeling task.

Chapter 3: Details the integration of the forceps device into the Vitreoretinal SmartArm system, introducing new capabilities such as rotation and grasping. This chapter outlines the enhancements made to the hardware architecture, software systems, and kinematic framework to accommodate the forceps' unique requirements.

Chapter 4: Focuses on refining the image processing pipeline and adapting the shadow-based positioning framework for the forceps device. It introduces advancements, such as the integration of specialized neural networks (NN) for keypoint detection, and concludes with an experiment designed to validate the system's

effectiveness in forceps-based autonomous positioning tasks.

Chapter 5: Presents the experimental results and analysis of the shadow-based autonomous positioning with the forceps device. This chapter provides insights into the system's performance, identifies key challenges, and offers a critical assessment of its strengths and limitations.

Chapter 6: Proposes a novel forceps alignment method to enhance safety and precision during ILM peeling. Additionally, this chapter outlines a structured experimental plan to evaluate the robotic system's feasibility for performing ILM peeling in a controlled environment.

Chapter 7: Summarizes the key findings of this research, highlights its contributions to robotic vitreoretinal surgery, and provides actionable recommendations for future research and system development.

Chapter 2 Background and System Overview

This chapter provides a comprehensive overview of the field of robotics, with a focus on the evolution and application of surgical robotic systems. It begins with a brief history of robotics and examines the current surgical robots in use, highlighting their capabilities and the advancements made toward achieving higher levels of autonomy. The discussion then transitions to the Vitreoretinal SmartArm system, the robotic platform central to this research. Its design, capabilities, previous applications, and identified limitations are explored in detail. The limitations highlight the motivation behind the research in this thesis. Finally, the chapter delves into the internal limiting membrane (ILM) peeling task, outlining its significance in vitreoretinal surgery and the unique challenges it presents.

2.1 Robotic Systems

2.1.1 History of Surgical Robotic Systems

Although robots have long been used in industrial settings for their precision with repetitive tasks, their integration into the medical field has been a more recent development. Surgical robotics can be categorized into three types: active, semi-active, and leader-follower (or master-slave) systems. Active systems operate autonomously with surgeon oversight, semi-active systems blend pre-programmed functions with surgeon control, and leader-follower

systems, such as the da Vinci® and ZEUS platforms, directly replicate the surgeon’s hand movements within the patient’s body.

Traditionally, surgical procedures required large incisions to access internal organs, often resulting in significant pain, high infection rates, considerable blood loss, and extended hospital stays. During the mid-nineteenth century marked the advent of modern surgical practices with the introduction of anesthesia and antiseptic techniques, greatly improving patient outcomes. By the mid-1970s, Minimally Invasive Surgery (MIS) emerged as a favored alternative to open surgery, utilizing multiple small incisions instead of single large ones which offers substantial benefits such as reduced pain, smaller scars, and quicker recovery times [13]. However, early MIS procedures were challenging due to limited visibility, resulting in longer surgeries and a higher risk of orthopedic injuries for surgeons.

The 1980s brought significant advancements in imaging technology, such as solid-state cameras and high-definition video displays, allowing surgeons to view the patient’s anatomy in high-quality 3D images during procedures. These innovations resolved visibility issues and led to an increased preference for laparoscopic techniques over traditional open surgery, resulting to lower postoperative complications and better cosmetic outcomes.

Robot-Assisted Surgery (RAS) began to develop in the 1980s with systems like the Arthrobot, which were used for prostatectomies and cardiac valve repairs. RAS aims to overcome the limitations of traditional MIS by enhancing surgical precision and control. The concept of Minimally Invasive Flexible Surgery (MIFS), which involves using devices attached to a laparoscope for both diagnosis and therapy, has also emerged from this technological evolution.

A timeline of the advancements of surgery with the integration of robots over time [14] is depicted in Figure 2.1.

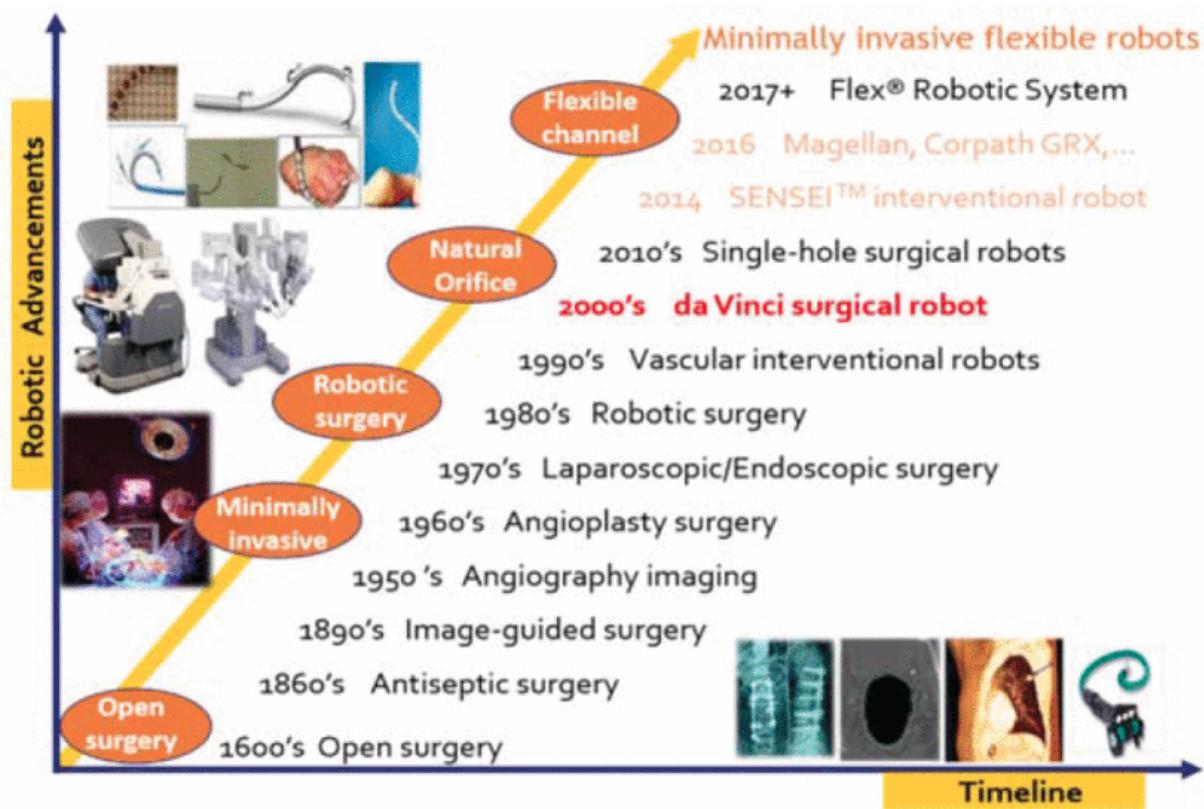


Fig. 2.1: Roadmap of the advancement of surgery with robot integration over time [14].

2.1.2 Current Surgical Robotic Systems

Leader-follower teleoperation systems, such as the da Vinci® and hinotori™ surgical systems, have been pivotal in the development of robotic surgery. In these systems, surgeons operate from a console located outside the surgical room, issuing control commands to robotic end effectors. Visual and haptic feedback provides real-time information, allowing surgeons to perform procedures with high accuracy and minimal direct contact with patients. The da Vinci® surgical system, developed by Intuitive Surgical® (Sunnyvale, CA, USA), is one of the most notable and widely used in this field. Since its FDA approval in 2000, the da Vinci® system has undergone continuous advancements and has been utilized in over 12 million procedures globally as of 2022 [15].

The da Vinci® system consists of a patient-side cart, which houses the robotic manipulators, and a surgeon console equipped with two input devices and a stereo high-definition screen. The console also includes pedals, the most critical of which is the clutch, allowing the surgeon to activate or deactivate communication between the input devices and the robotic manipulators. This setup enables the surgeon to control the robotic instruments with great precision, benefiting from features such as motion scaling and hand tremor cancellation. The da Vinci® system also incorporates gravity compensation algorithms to prevent the operator from feeling the weight of the input devices, enhancing ergonomic comfort during lengthy procedures.



Fig. 2.2: The da Vinci® 5 video cart (left), the da Vinci® 5 surgeon console (middle), and the da Vinci® 5 patient side cart (right). (Images® 2024 Intuitive Surgical, Inc).

The latest and most advanced system, the da Vinci® 5, includes over 150 enhancements compared to the da Vinci® Xi [16]. The accuracy and precision have been improved from the older models (S, Si, and Xi), which had a reported accuracy of 1 mm^{*1} [17] [18]. The computing and image processing power has increased more than 10,000 times compared to the Xi. One of the most notable advancements is the implementation of Force Feedback technology,

^{*1}Only reported by Intuitive Surgical® as no research has compared the 5 to older models

which allows surgeons to feel the forces exerted on tissue during surgery, resulting in 43% less force applied to the tissue. Additionally, the console ergonomics have been improved, enabling surgeons to adjust the viewing position for better posture and comfort.

In addition to the da Vinci® system, the hinotori™ Surgical Robot System by Medicaroid Corporation, as shown in Figure 2.3, has emerged as a competitive alternative [19]. Developed as a joint venture between Kawasaki Heavy Industries and Sysmex Corporation, the hinotori™ system received regulatory approval in Japan in 2020 and has demonstrated performance comparable to the da Vinci® system. It offers a cost-effective solution, with an anticipated price significantly lower than that of the da Vinci®, making advanced robotic surgery more accessible.

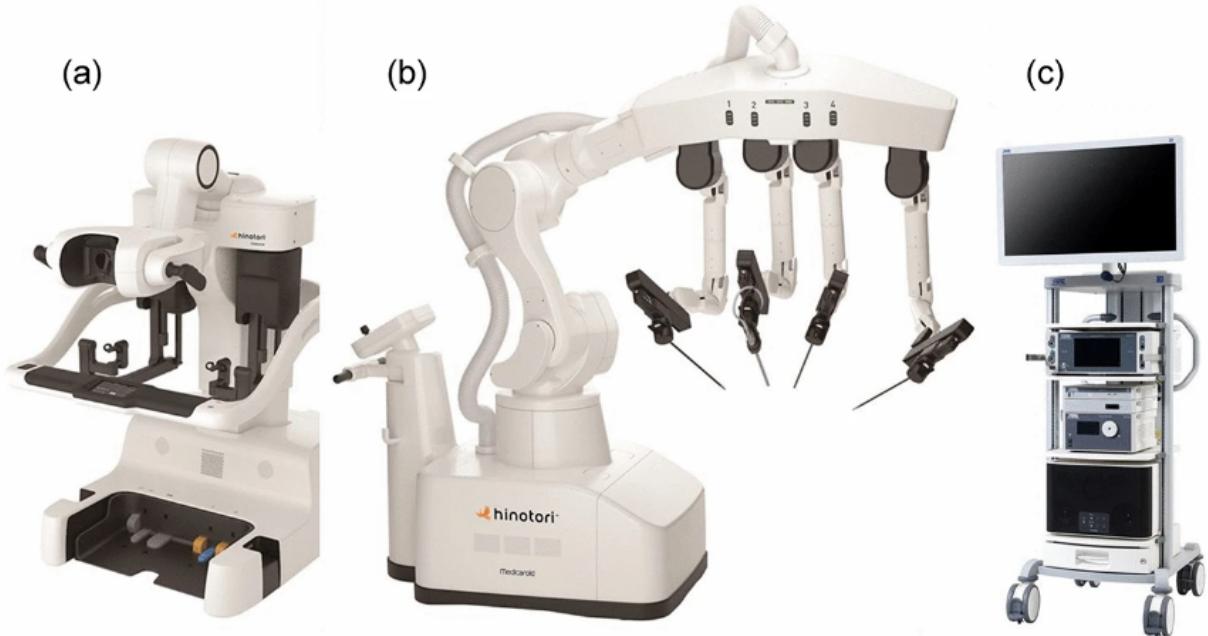


Fig. 2.3: The hinotori™ surgeon cockpit (a), the operation robot unit (b), and monitor cart (c) (Images © 2022 Medicaroid Corporation).

The Senhance® Surgical Robotic System by Asensus Surgical, as shown in Figure 2.4 presents a different approach by replicating the operation mode of manual laparoscopy [20]. This system retains the fulcrum effect, making it easier for surgeons experienced in manual

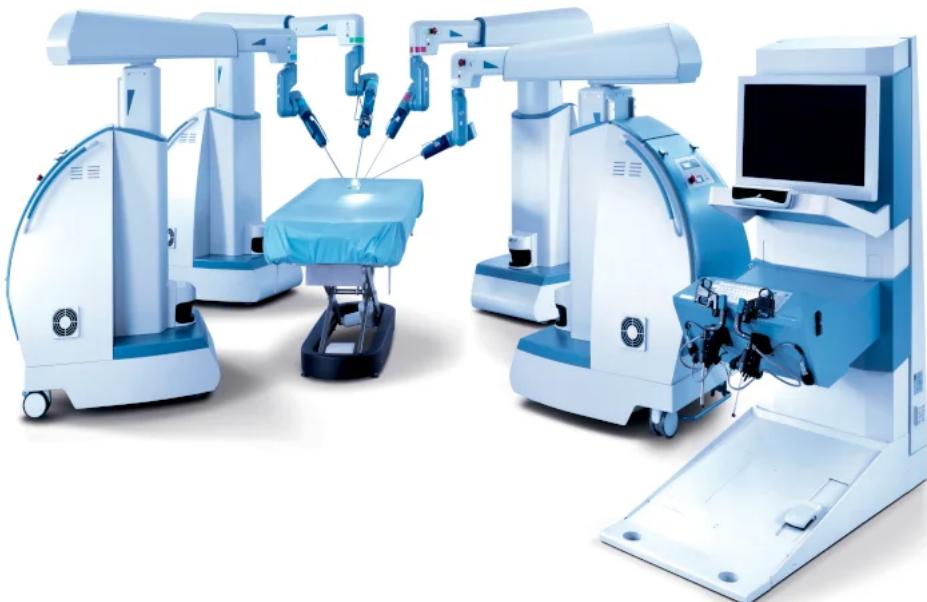


Fig. 2.4: The Senhance® Surgical Robotic System (Images © 2024 Asensus Surgical USA, Inc).

laparoscopy to transition to robotic surgery. Similar to the other systems, the Senhance® system enhances the surgeon’s capabilities with features like motion scaling and hand tremor cancellation.

Microsurgery also benefits greatly from robotic assistance. In ophthalmology, the PRECEYES Surgical System R1.1, as shown in Figure 2.5, offers precise manipulation for vitreoretinal procedures [21]. This system attaches surgical instruments to a robotic arm fixed to the operating bed, which the surgeon controls with a joystick. The system filters hand tremors and scales motion, providing highly accurate control. The PRECEYES system has demonstrated precision better than $20 \mu\text{m}$, significantly improving outcomes in delicate eye surgeries. The PRECEYES system is the first clinically validated vitreoretinal robot system for use that has a CE mark and is commercially available.

Additionally, the Luca™ surgical robotic system by AcuSurgical represents a significant advancement in robotic eye surgery as a bi-manual/dual-instrument robot designed for oph-

thalmic procedures [22]. In a recent first-in-human clinical study, the system successfully performed core vitrectomy procedures on all seven patients, demonstrating its effectiveness and reliability (Figure 2.6). With a precision of up to $10\text{ }\mu\text{m}$, the LucaTM system enables a wider range of pathologies to be addressed with greater safety and precision. Although still under development and not yet commercially available, the LucaTM holds great promise for expanding the capabilities of robotic-assisted eye surgery.

Similarly, the MUSA-3 system by MicroSure, as shown in Figure 2.7 supports lymphatic and hand surgeries. This system is compatible with conventional surgical instruments and microscopes, making the transition to robotic assistance seamless for surgeons. The surgeon operates the robotic arms via joysticks and a large screen using a digital or hybrid microscope. The MUSA filters hand tremors and scales down input motions, allowing for precise operations such as anastomosis of lymphatic vessels smaller than 0.3 mm [23].

The international adoption of surgical robots is driven by their ability to enhance MIS and other surgical procedures. The separation of the patient side and the surgeon console in leader-follower systems allows for the integration of various functionalities, such as motion scaling and tremor cancellation, which improve surgical precision and reduce risks. These advancements in robotic surgery are paving the way for safer, more efficient, and more effective surgical procedures across multiple medical fields.

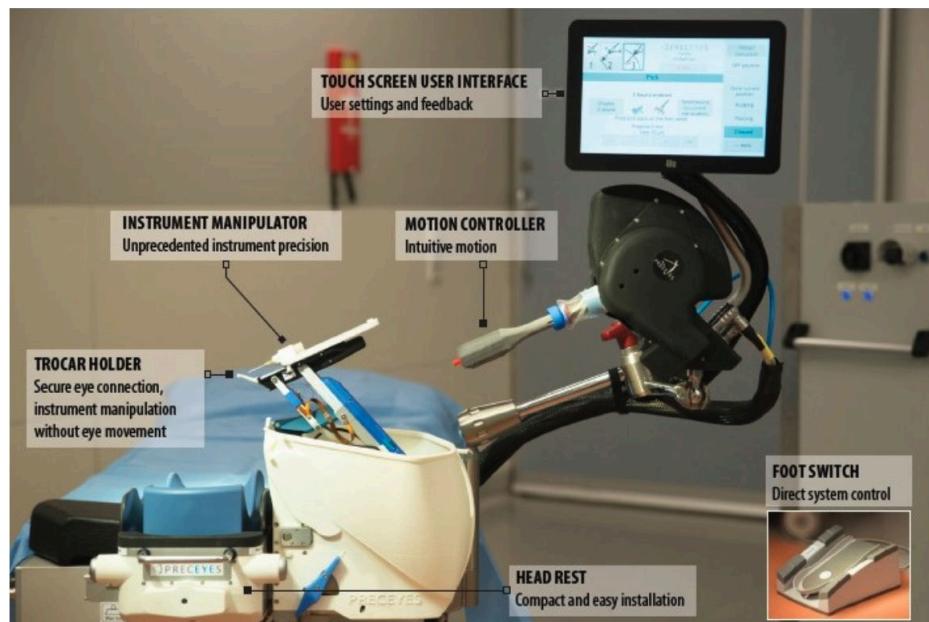


Fig. 2.5: The PRECEYES Surgical System R1.1 (Images © 2024 PRECEYES B.V.).

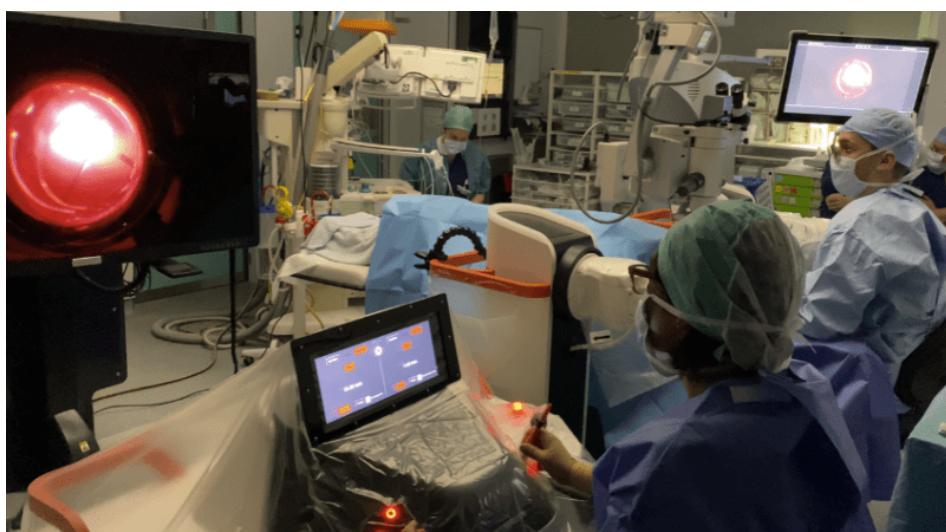


Fig. 2.6: The Luca™ Surgical Robotic System being used during clinical trial (Images © 2024 AcuSurgical SAS).



Fig. 2.7: The MUSA-3 robotic arm cart (left) and surgeon console (right) (Images © 2024 Microsure NL).

2.1.3 SmartArm Surgical Robotic System

Our group^{*2} has been actively developing surgical robotic systems to address various challenges in the fields of microsurgery and minimally invasive surgery. All of our robotic systems—including the pediatric system [24], moonshot system [25], and vitreoretinal system [26,27]—are currently based on the SmartArm surgical robotic platform [28].

The SmartArm platform was specifically developed to address several key challenges identified both within the industry and by other research groups. While the da Vinci® system has been highly successful in larger workspaces, it encounters difficulties when operating in narrow or confined spaces. This limitation is partly due to its mechanically static Remote Center of Motion (RCM), the fixed point where surgical instruments enter the body. Additionally, the da Vinci® system's relatively large size makes it less suitable for navigating complex and restricted environments.

^{*2}<https://sites.google.com/g.ecc.u-tokyo.ac.jp/cdbim-medical-devices>

In contrast, the SmartArm system offers greater versatility with its software-based RCM, allowing the robotic platform to be adapted not only for various surgical operations but also for applications in other fields. The platform’s methods are grounded in a mathematical framework that is specifically tailored to perform a variety of tasks effectively. Another significant advantage of the SmartArm system is its capability for limited autonomy. Current surgical robotic systems, like the da Vinci®, provide little to no autonomy, as they are entirely controlled by surgeons due to concerns about liability and safety. However, in academic research, automation is a heavily explored area, where safety concerns are less restrictive, allowing for greater innovation in robotic autonomy.

The Level of Autonomy (LoA) can be classified into six stages, as outlined in Figure 2.8, according to the ISO/IEC standardization framework [29]. At LoA 0, the surgeon has complete control over the entire procedure, which is where systems like the da Vinci® fall. On the opposite end of the spectrum, LoA 5 represents full autonomy, where the robot independently creates and executes the procedure—this is the ultimate goal for robotic autonomy.

Several studies have focused on reducing the burden on surgeons by enabling robotic systems to operate partially autonomously, especially in environments lacking perception capabilities, where managing complex robotic motions is critical [30, 31]. LoA 4, which involves automating specific tasks under certain conditions, is a common focus of current research [32, 33]. The concept of virtual fixtures is also extensively explored, primarily to restrict the robot’s motion to a safe area, thereby providing it with some autonomy to avoid obstacles on its own, typically falling under LoA 2 or 3 [34, 35].

However, kinematically redundant robotic systems, where the robot’s end posture can be achieved through multiple joint configurations, require a higher level of autonomy to prevent collisions that the surgeon cannot directly control. In the SmartArm system, the implementation of active constraints using Vector Field Inequalities (VFI) allows the robot to autonomously avoid obstacles, creating a robust platform for advancing research in robotic autonomy, equivalent to LoA 1 [36]. More information on VFI’s can be found in Section 8.4.

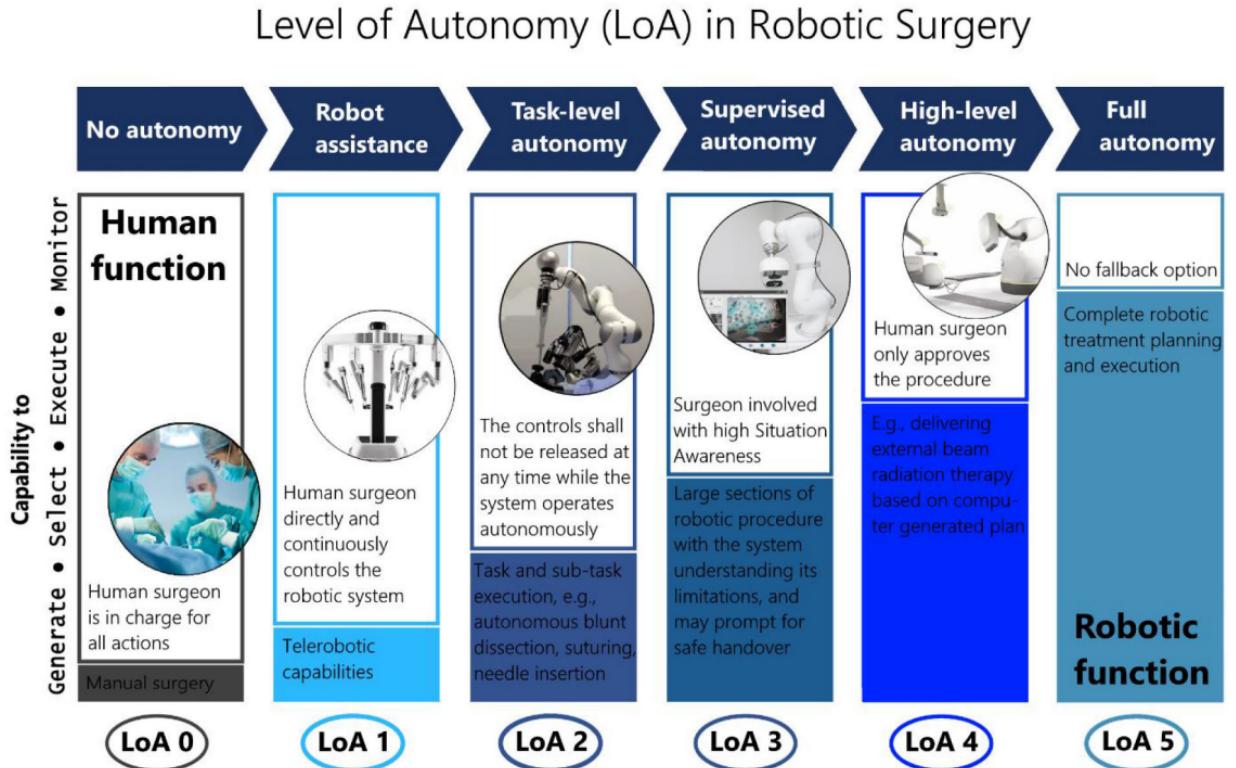


Fig. 2.8: The 6-stage classification for assessing the autonomous capabilities of surgical robots [29].

2.2 Vitreoretinal SmartArm System

The SmartArm system specialized in vitreoretinal surgery is used throughout this research. The following subsection describe the hardware and software architecture in detail, as well as previous research performed with this system [12]. Furthermore, the limitations of the current system is discussed to set the foundation for improvements mentioned later in this thesis.

The previous research on the Vitreoretinal SmartArm system has laid a strong foundation for future advancements. This work not only benefits a wide range of ophthalmic surgeries but also demonstrates the feasibility and effectiveness of several critical components. The shadow-based autonomous positioning establishes a reliable pipeline for detecting the instrument positions within the workspace. The implementation of task constraints enhances both

the safety and usability of the system. Additionally, the automated light guide provides optimal illumination of the primary surgical instrument, thereby reducing the surgeon's workload. Lastly, orbital manipulation expands the microscopic workspace, paving the way for new surgical techniques to be developed and applied.

As discussed in Section 2.1.3, the SmartArm system operates LoA 1 by utilizing VFI's to maintain the Remote Center of Motion (RCM), ensuring that the robot precisely matches the patient's eye insertion points. The underlying mathematical concepts, including dual quaternions, Jacobians, distance functions, the constrained optimization algorithm, and VFIs, are thoroughly explained in the Appendix.

2.2.1 Hardware and Software Architecture of Previous Work

Hardware Architecture

The Vitreoretinal SmartArm system is a teleoperated system composed of a patient side and operator side as illustrated in Figure 2.9. The information between the robots and computers is exchanged via Ethernet on an isolated Local Area Network (LAN). The patient side consists of two follower robot arms (DENSO VS050^{*3}, DENSO WAVE Inc., Japan) that are attached to their respective robot controllers (DENSO RC8^{*4}, DENSO WAVE Inc., Japan). The primary follower robot has the dominant surgical instrument, to perform the task, such a surgical needle (Flomax ^{*5}, Nipro, Japan), used in previous research. The other follower robot has a light guide (Straight Endoilluminator ^{*6}, Alcon, USA), to illuminate the workspace. The phantom eye model (Bionic-Eye ^{*7}, Mitsui Chemical, Japan), a surgical practice tool to replace pig eyes, is utilized for the various procedures and experiments replicating a realistic workspace

^{*3}<https://www.denso-wave.com/ja/robot/product/five-six/vs050-060.html>

^{*4}<https://www.denso-wave.com/en/robot/product/controller/rc8a.html>

^{*5}https://med.nipro.co.jp/med_eq_category

^{*6}<https://alcon.widen.net/s/zrlg779yi0>

^{*7}<https://jp.mitsuichemicals.com/jp/special/bionic-eye/project.htm>

for development. Trocars (25 Ga Cannula *⁸, D.O.R.C., Netherlands) are inserted into the eye to allow the surgical instrument, light guide, and irrigation to be inserted. A disposable flat lens (HHV Dispo type 5d *⁹, HOYA Corpo-ration, Japan) is placed on the Bionic-Eye to correctly adjust the view of the workspace. A digital ophthalmic microscope (OPMI Pico *¹⁰, ZEISS, Germany) with a 4K camera (STC-HD853HDMI*¹¹, Omron-Sentech, Japan) is utilized to allow teleoperation from afar.

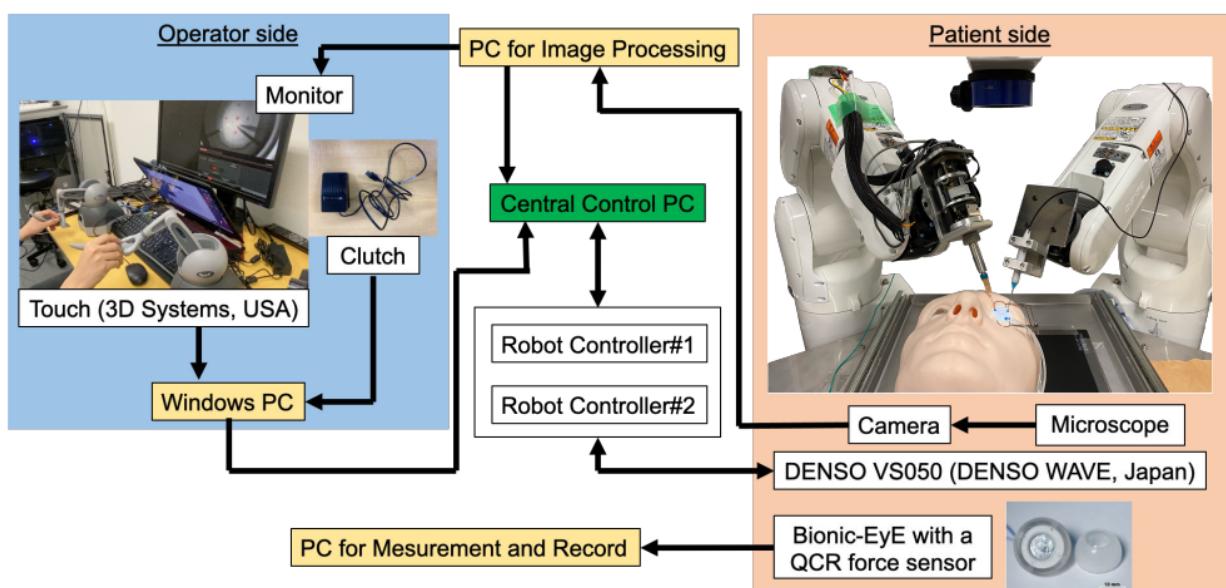


Fig. 2.9: The Hardware Architecture of the Vitreoretinal SmartArm system. A central control PC manages the operator and patient sides of the system [12].

The operator side consists of a PC, equipped with an NVIDIA GeForce RTX 3090 GPU (now 3070 GPU), used for microscopic image capturing and processing. The image is captured via digital microscope 4K camera setup connected to a capture board (DeckLink Quad HDMI Recorder *¹², Blackmagic Design, Australia) which is also displayed to the operator via monitor. The required robot control information is sent to the central control PC. Two

*⁸<https://dorcglobal.com/product/25g-cannula-set-high-flow-infusion-line>

*⁹https://jp.hoyasurgicaloptics.com/down/uploads/2022_698_1646659438.pdf

*¹⁰<https://www.zeiss.com/meditec/en/products/surgical-microscopes/opmi-pico.html>

*¹¹<https://sentech.co.jp/en/products/DVI-SDI/4K.html#cnt2nd>

*¹²<https://www.blackmagicdesign.com/jp/products/decklink/techspecs/W-DLK-36>

leader haptic interface devices (Touch ^{*13}, 3D Systems, USA) and the clutch are managed by a separate Windows PC and are utilized to manipulate the follower robots. The motion scaling ratio is set at 1/30, significantly reducing the difficulties of precision such as hand tremors. The central control PC utilizes an NVIDIA GeForce RTX 3070 GPU (now Titan GPU) to process the current and desired joint positions of the follow robots.

Software Architecture

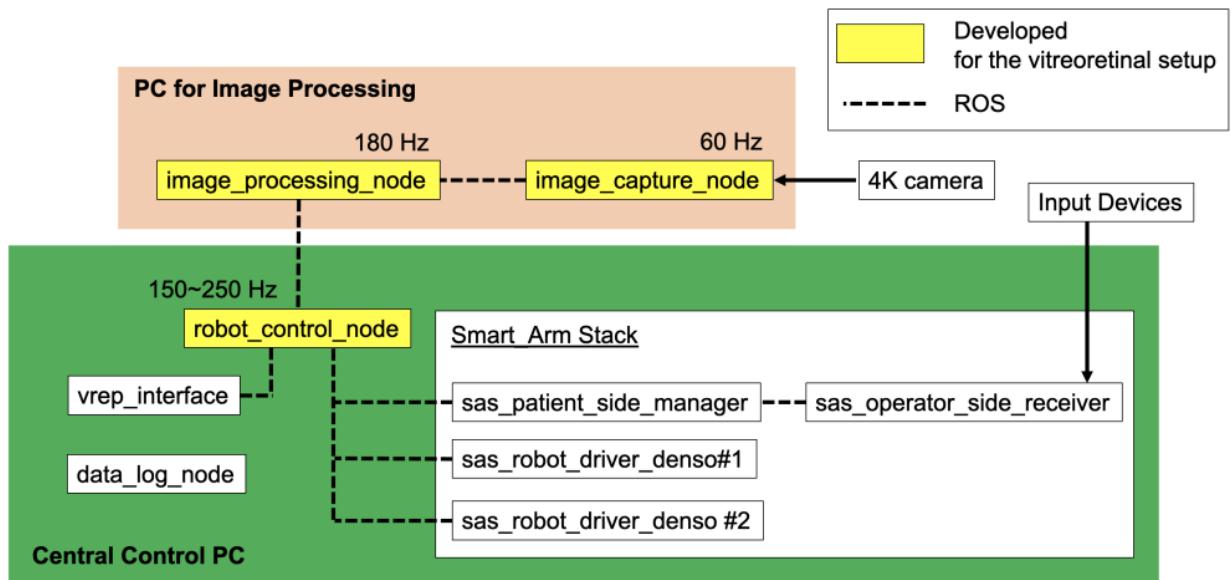


Fig. 2.10: The Software Architecture of the Vitreoretinal SmartArm system [12].

The software architecture of the system composes of an image processing and central control PC as shown in Figure 2.10. Both computers operate on a supported Ubuntu distribution (Ubuntu 20.04 LTS, Focal Fossa ^{*14}). These systems utilize the Robot Operating System (ROS Noetic Ninjemys ^{*15}) for inter-process communication and coordination. ROS, an open-source framework, facilitates seamless integration and communication between the

^{*13}<https://tinyurl.com/ms2y64ht>

^{*14}<https://releases.ubuntu.com/focal/>

^{*15}<http://wiki.ros.org/noetic>

various components of the robotic system, ensuring efficient operation and control. One of the core features of ROS is its use of nodes, which are individual processes that perform specific tasks within a robot's system. These nodes communicate with each other through a standardized messaging system, allowing for a modular and scalable architecture. This design enables developers to break down complex robotic systems into smaller, manageable components, each running as a separate node. By using nodes, developers can easily add, remove, or update parts of the system without affecting the overall functionality. This modularity not only simplifies the development process but also enhances system robustness and flexibility, making ROS a powerful tool for building sophisticated robotic applications.

2.2.2 Shadow-Based Autonomous Positioning

The shadow-based autonomous positioning method developed in prior work [12, 26] has demonstrated significant potential for enhancing surgical precision by leveraging real-time keypoint detection of the surgical needle and its shadow. The method's strategy is organized into four steps, as illustrated in Figure 2.11 and detailed below:

1. Planar Positioning:

- The surgical instrument's tip is first moved parallel to the image plane to a safe point above the target point.
- This step converges when the distance between the surgical instrument's tip and the target point, d_{planar} , becomes less than or equal to 20 pixels (approximately 50 μm) in the microscopic view.

2. Overlap Positioning:

- This step ensures that the instrument's tip and its shadow do not overlap, a critical requirement for the subsequent vertical positioning.
- The step completes when the distance between the shadow's tip and the surgical instrument's shaft, d_{shaft} , becomes greater than or equal to 100 pixels (approxi-

mately 240 μm) in the microscopic view.

3. Vertical Positioning:

- The surgical instrument's tip is moved vertically to approach the target point on the retina.
- This step converges when the distance between the instrument's tip and its shadow's tip, d_{tip} , becomes less than or equal to 50 pixels (approximately 120 μm) in the microscopic view.

4. Additional Positioning:

- After the vertical positioning step, the surgical instrument's tip has not yet reached the retina. To ensure proper contact, the controller moves the instrument downward by an additional distance.
- The remaining distance, d_{rest} , is calculated using a geometrical relationship (no image processing involved):

$$d_{rest} [\mu\text{m}] = 50 [\text{px}] \times \frac{dz [\mu\text{m}]}{d_{xy} [\mu\text{m}]} \times \frac{1000}{417 [\text{px/mm}]} \quad (2.2.1)$$

where d_{xy} and d_z are the xy and z components of the relative position between the light guide's tip and the instrument's tip, respectively.

- To achieve precise positioning, the controller moves the instrument downward by:

$$d_{add} = d_{rest} + 130 \mu\text{m} \quad (2.2.2)$$

- The value of 130 μm was determined through a preliminary experiment. In this experiment, after the vertical positioning step, the surgical instrument was moved downward until its tip touched the retina. The mean difference between the actual distance required to reach the retina and d_{rest} was calculated as 50 μm , with a standard deviation of 80 μm .

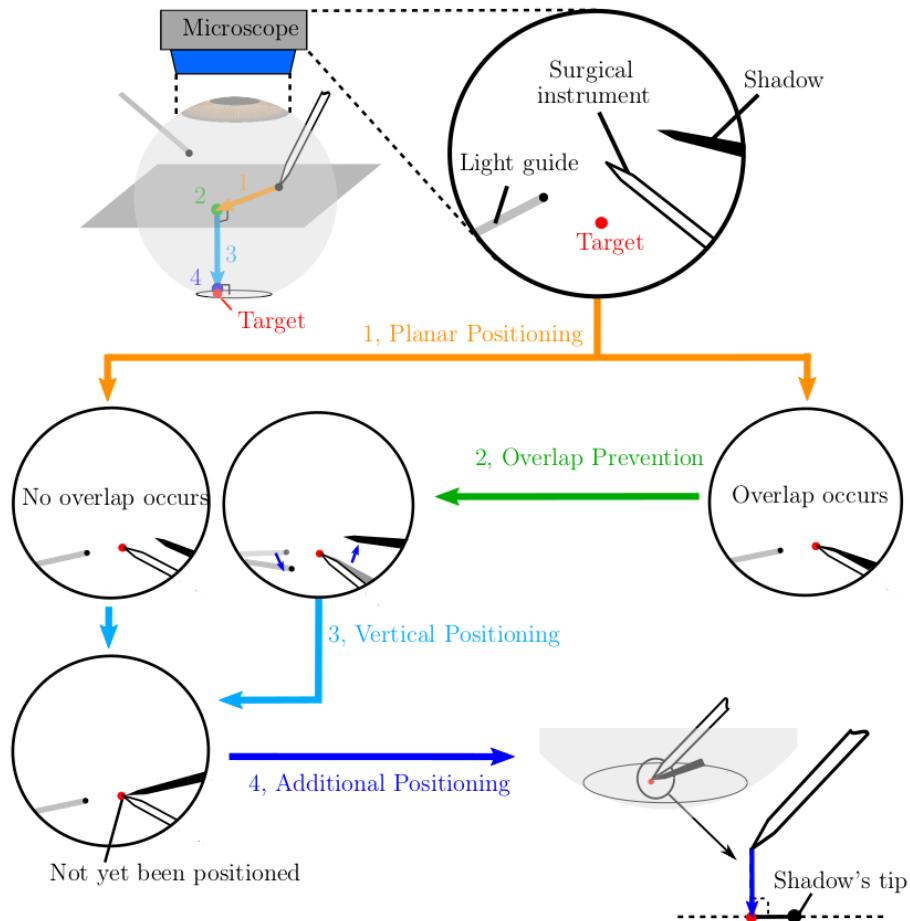


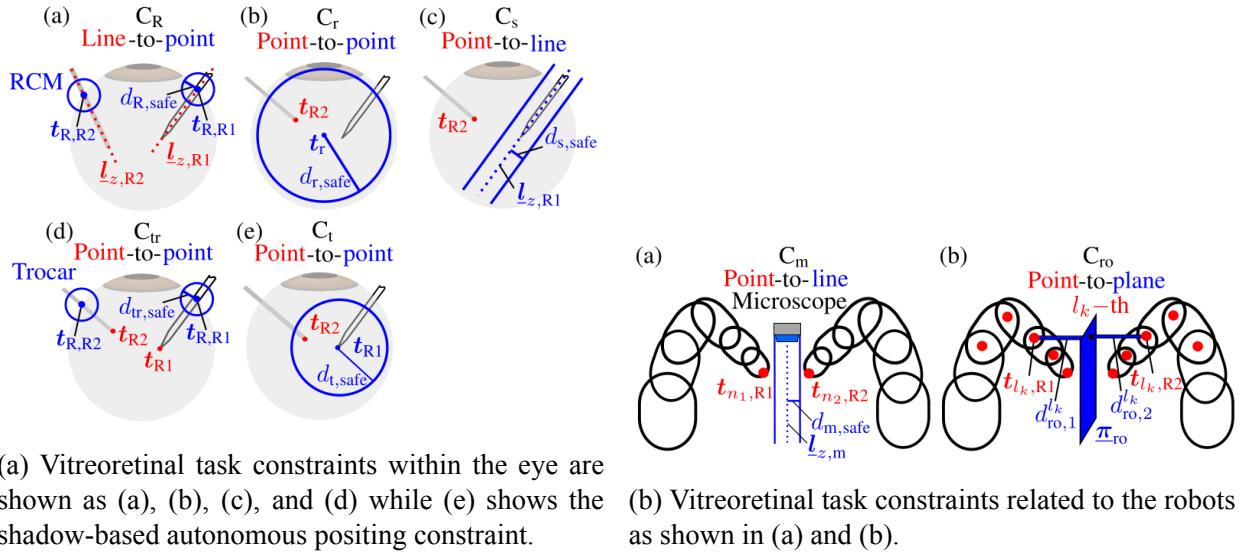
Fig. 2.11: Visualized four-step shadow-based autonomous positioning strategy [26].

2.2.3 Vitreoretinal Task Constraints

The following task constraints are implemented into the Vitreoretinal SmartArm system to safely conduct the vitreoretinal tasks by preventing collisions and keeping the desired points within the workspace as also shown in Figure 2.12:

- C_R : The instrument shafts must consistently pass through their designated insertion points (RCM) along the edge of the eye.
- C_r : The light guide tip must avoid contact with the retina.
- C_s : The instrument shafts must avoid colliding with each other .

- C_{tr} : The instrument tips must always stay within the eye.
- C_m : The robots must prevent collisions with both the microscope.
- C_{ro} : The robots must prevent collisions with each other.
- C_j : The joint values must stay within their specified limits.



(a) Vitreoretinal task constraints within the eye are shown as (a), (b), (c), and (d) while (e) shows the shadow-based autonomous posing constraint.

(b) Vitreoretinal task constraints related to the robots as shown in (a) and (b).

Fig. 2.12: The geometrical primitives and types of VFI's used to generate each task constraint in the Vitreoretinal SmartArm system [26].

To ensure the effectiveness of shadow-based positioning and maintain the visibility of the shadow, three additional constraints are imposed:

- C_1 : The tip of the instrument's shadow must remain visible in the microscopic view at all times.
- C_t : The tip of the instrument must be sufficiently illuminated.
- C_2 : The contour of the instrument's shadow must be clearly visible throughout the procedure.

Conical constraints are applied to ensure that conditions C_1 and C_2 are met, guaranteeing the visibility of the surgical instrument's shadow within the circular microscopic view as shown in 2.13. Geometrically, the shadow's tip remains visible through the microscope as long as the

surgical instrument's tip stays within the cone formed by the light guide's tip and the circular microscopic view. The surgical instrument's tip is properly illuminated as long as it remains within the illumination volume of the light guide, which also takes the shape of a cone. C_t is generated with a point-point Jacobian as shown in Figure 2.12-(e). The combination of the conical constraints enable the light guide to become (semi-)autonomous.

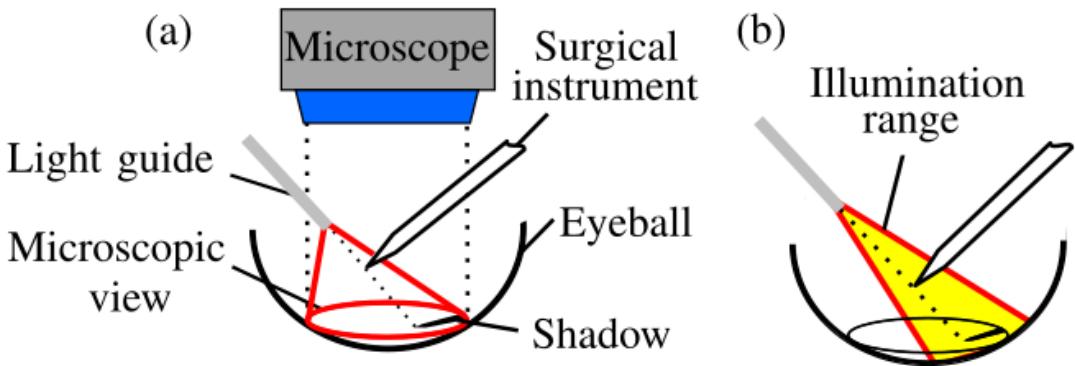


Fig. 2.13: (a) The instrument shadow tip remains visible when the instrument's tip is within the light guide cone and microscope view. (b) The instrument shadow tip is properly illuminated when it remains within the illumination cone [26].

2.2.4 Image Processing the Instrument Tip and its Shadow

To detect the the tip's of both the instrument and its shadow, a UNet-based image-processing strategy was employed. There exists an older segmentation model as discussed in [26], however, the more recent strategy will be discussed. The neural network (NN) for vitreoretinal keypoint detection was trained and optimized to detect a surgical needle and its shadow within the image workspace. A region of interest (ROI) bounding box was utilized to focus on a smaller, more detailed area containing the surgical needle and shadow. Within the ROI, keypoints such as the needle's tip, the needle's shadow, and the instrument shaft were detected. For successful initial detection, the instrument and its shadow needed to be positioned within the ROI.

The ROI's size and coordinates were dynamically adjusted based on the previously detected

coordinates of the needle and shadow. Specifically, the center of the ROI was determined by the midpoint of the detected needle and shadow, while the ROI’s size decreased as the needle tip and shadow moved closer together and increased when they moved apart. This adaptive ROI ensured an accurate and efficient keypoint detection for its application.

The NN was built using a ResNet-18 encoder pre-trained with ImageNet weights, providing a robust foundation for surgical instrument detection. Training utilized a dataset of 600 microscopic images, expanded to a total of 2500 images through image augmentation. 250 images for validation, and 250 images for testing were used. The network was trained using Mean Squared Error (MSE) loss, Adam optimizer with a learning rate of 0.0001, and batch size of 5. The MSE loss on the test set was 2.127×10^{-4} . The image processing was capable of updating the distance at a rate of 60Hz. The network did not rely on preimage processing prior to detection, simplifying the pipeline.

The image capturing node is a ROS node that handles the video frames from the microscope 4k camera, captures the target points, applies the keypoint detection NN to output the tip location within the workspace, and displays a marked up image of the workspace and closeup of the ROI. The markups include important information such as the ROI, keypoints, shaft line, planar error, shaft distance, tip distance, ROI size, current step, and if the tip is in contact with the retinal surface as illustrated in Figure 2.14.

2.2.5 Comparing Different Levels of Autonomy

Three experiments were conducted to evaluate the performance of different levels of autonomy in a five-point positioning task: manual operation (**MO**), teleoperation (**TO**), shared autonomy (**SA**), and full autonomy (**FA**) [12]. The experiments measured the contact and force exerted by a surgical needle on the surface of an eye fundus. To achieve this, an Arduino-based system was used to detect circuit continuity for contact detection, while a quartz crystal resonator (QCR) force sensor measured the applied force. The experimental setup is illustrated

in Figure 2.15.

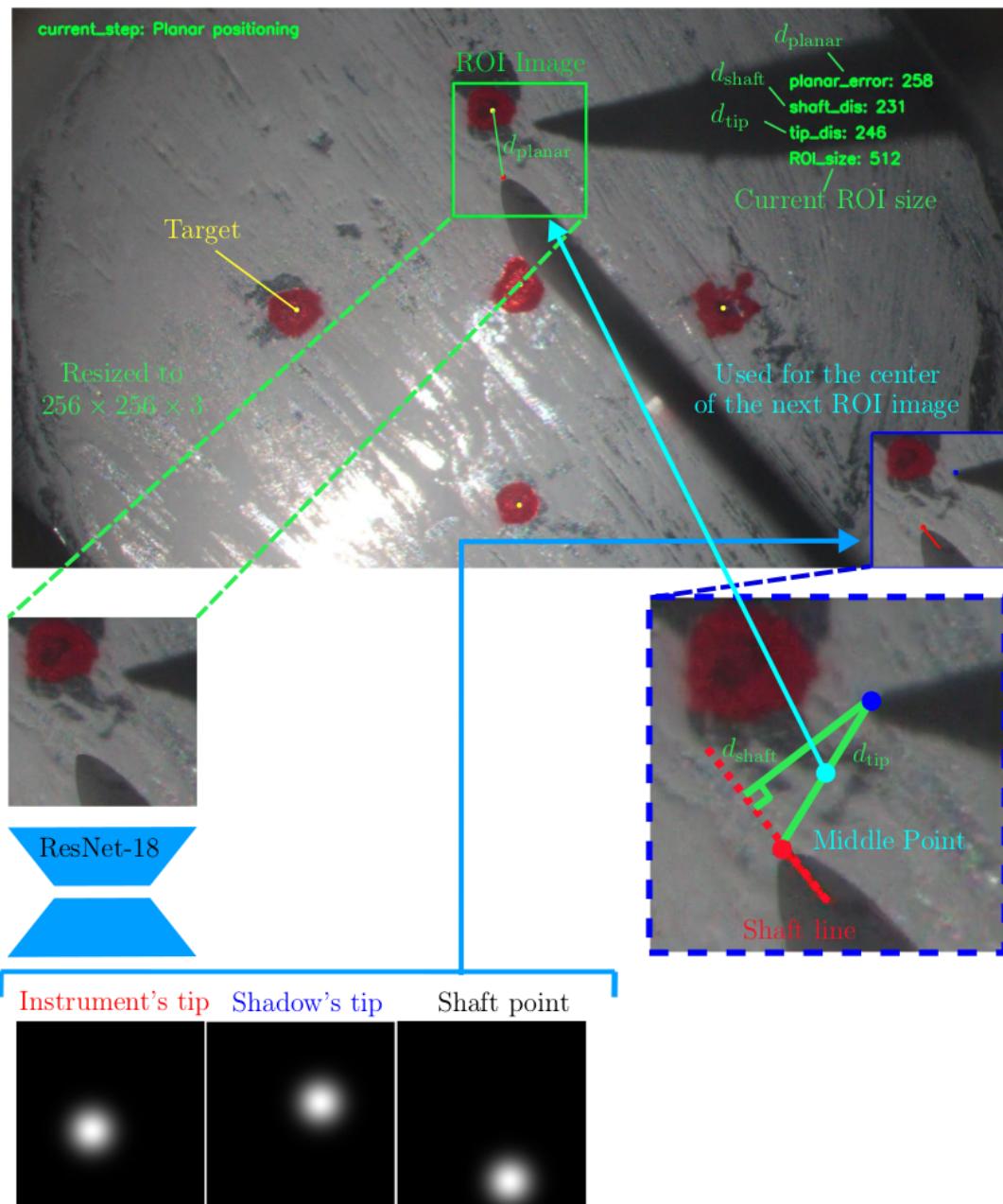


Fig. 2.14: The marked image displayed to the user during the shadow-based autonomous positioning procedure [12].

Evaluation of Usability by Expert Surgeons

Six surgeons from the University of Tokyo Hospital, Japan, participated in the experiment.

Each surgeon performed five trials under the **MO**, **TO**, and **SA** configurations, resulting in a total of 15 trials per surgeon. The median results from these trials are presented in Table 2.1.

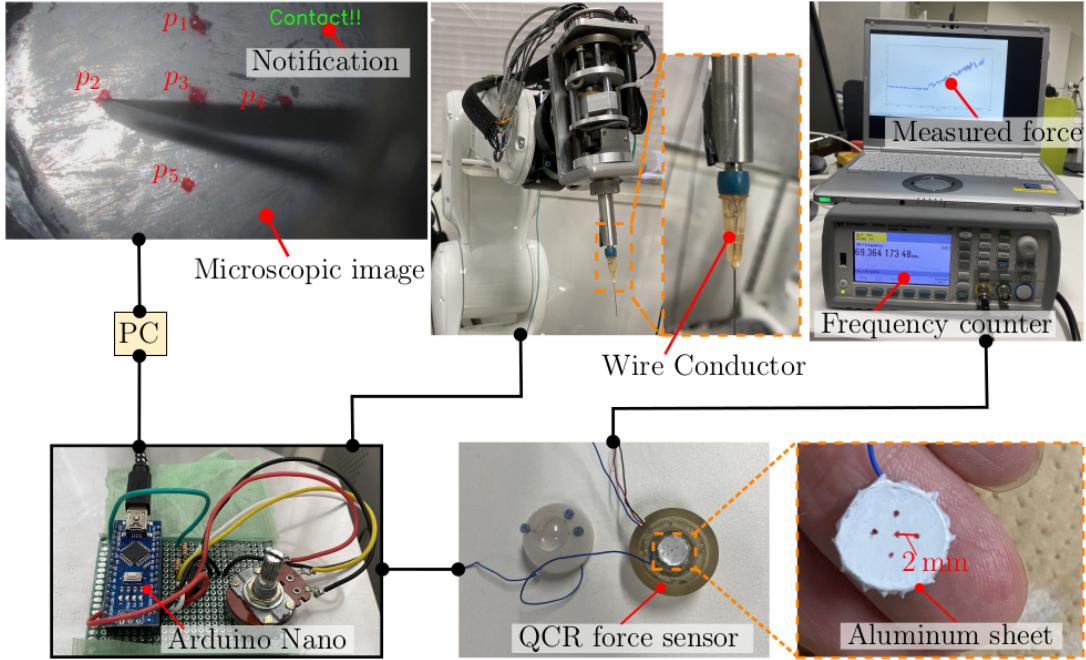


Fig. 2.15: Experimental setup using an Arduino system for contact detection and a QCR force sensor for force measurement [12].

Key findings include:

- **Positioning Accuracy:** The **SA** configuration demonstrated improved positioning accuracy compared to **MO** and **TO**, as determined by a post-hoc Dunnett's test.
- **Workload Reduction:** NASA-TLX metrics suggest that **SA** reduced workload across all measured dimensions.
- **Force Comparison:** The maximum force applied in **MO** was lower than in **SA**, and both were significantly lower than in **TO**.

Evaluation of Autonomous Positioning

The **FA** experiment utilized the shadow-based autonomous positioning method to perform the same five-point positioning task evaluated by the expert surgeons. Out of seven trials, five were completed without force quitting, and the results are shown in Table 2.2.

Surgeon #	Error [$\times 10^2 \mu\text{m}$]			Workload		
	MO	TO	SA	MO	TO	SA
1	2.0	1.7	1.4	13.5	10.9	4.0
2	2.2	1.2	1.4	13.0	12.1	13.7
3	2.4	0.9	0.9	11.7	13.1	11.4
4	1.4	0.9	0.8	10.1	7.5	8.9
5	2.2	1.1	0.7	14.1	14.3	11.7
6	0.9	1.0	1.1	10.1	11.9	11.8
Median	2.0	1.0	1.0	12.4	12.0	11.5

Surgeon #	Ave. Force [mN]			Duration [s]		
	MO	TO	SA	MO	TO	SA
1	46	165	122	32	40	31
2	27	219	180	34	33	42
3	20	37	17	86	118	108
4	20	22	23	83	111	137
5	—*	84	52	31	38	41
6	23	17	15	57	103	150
Median	25	61	38	46	72	75

*The contact detection system did not correctly work.

Table 2.1: The median values of the five trials in each setup.

Key findings include:

- **Success Rate:** The needle tip reached the fundus in 68% of trials.
- **Task Duration:** The average task duration was 257 seconds, significantly longer than the 46 seconds recorded during the surgeons' manual operation.
- **Calibration Errors:** Positional errors between the robot and the microscope camera contributed to reduced accuracy and longer task durations.

These results suggest that reducing calibration errors and improving positional accuracy could significantly enhance the performance of fully autonomous positioning systems.

Evaluation of Novice's Learning Curve

	p_1	p_2	p_3	p_4	p_5
Trial1	310†	420*	330†	270†	400†
Trial2	280†	360*	300†	270†	400*
Trial3	310†	310*	—	—	—
Trial4	290†	430*	320†	270†	400†
Trial5	290†	—			
Trial6	270†	430*	330*	270†	410*
Trial7	290†	300†	290†	260*	360*

* not positioned, † positioned, —force-quit

Table 2.2: Results of autonomous positioning by FA.

To assess the learning curve for novice operators, twelve graduate student volunteers from the School of Engineering at the University of Tokyo were recruited. Each subject conducted the positioning task ten times, resulting in four subjects per trial. The results revealed significant findings through a post-hoc Tukey-Kramer test, including:

- **Force Safety:** Significant differences in average force were observed between **MO** and **TO**, suggesting that **TO** improved safety.
- **Task Duration:** Significant differences in task duration were noted between **TO** and **SA**, with **SA** reducing the time required for task completion.

These findings indicate that **TO** improves safety metrics for novices, while **SA** has the potential to reduce the challenge of extended robotic-assisted surgical tasks, facilitating smoother operation and reduced fatigue.

2.2.6 Orbital Manipulation

In ophthalmic surgery, particularly during tasks such as membrane peeling on the fundus, surgeons often intentionally adjust the RCMs of their instruments—a technique known as orbital manipulation. This method leverages the natural rotational freedom of the eye, allowing surgeons to access specific areas without needing to reposition the patient. Previous research conducted in our lab integrated these capabilities by integrating autonomous orbital manipu-

lation into the SmartArm system [27]. By enabling the eye to rotate autonomously in response to the movement of the surgical instruments, this system allows surgeons to perform vitreoretinal tasks within a significantly expanded workspace. The positioning with and without orbital manipulation is illustrated via Figure 2.16.

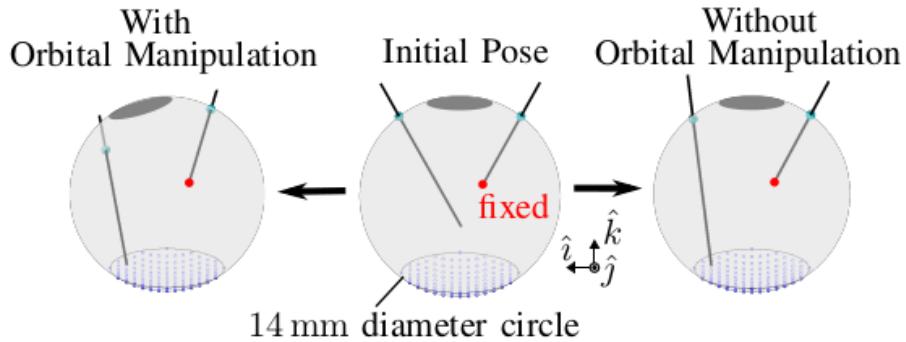


Fig. 2.16: Side view of positioning with and without orbital manipulation in simulation [27].

To advance this approach, a novel control strategy for orbital manipulation was proposed, including the derivation of a new distance function and its corresponding Jacobian to maintain the relative positions between RCMs using VFIs. Simulation and experimental results demonstrate that incorporating orbital manipulation not only enhances the system's manipulability but also allows the surgeon to observe a broader area around the fundus, thereby improving the overall effectiveness of certain surgical procedures. Comparison under a microscope view of using orbital manipulation and not can be observed via Figure 2.17.

2.3 Limitations of the Original System

The Vitreoretinal SmartArm system, originally designed for needle-based procedures, faced significant limitations that restricted its versatility and efficiency for advanced surgical tasks, such as ILM peeling. This section outlines the key challenges that motivated the redesign and enhancements introduced in this thesis, which are detailed in Chapters 3 and 4.

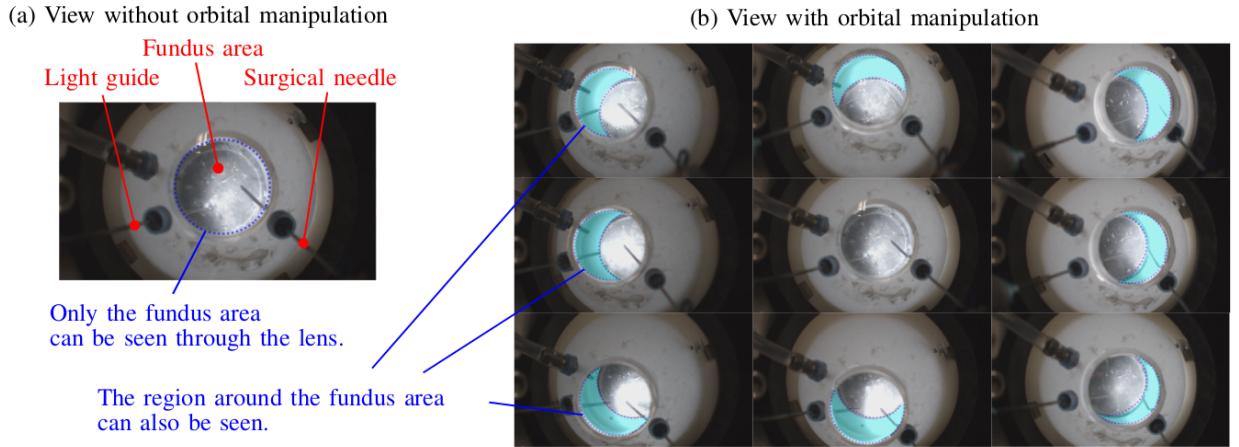


Fig. 2.17: Microscope view of positioning without (a) and with (b) orbital manipulation in the real system [27].

2.3.1 Challenges in Forceps Integration

The original system lacked the capability to support instruments with additional degrees of freedom, such as forceps. Unlike the needle, forceps require:

- **Rotational Freedom:** The ability to rotate independently without necessitating full robot repositioning.
- **Grasping Mechanism:** Precise control over opening and closing the tips, introducing new kinematic and control complexities.

These functional requirements highlighted the inadequacy of the existing kinematics and control framework, which was tailored for the needle's symmetric and fixed geometry. Incorporating the forceps demanded substantial updates to both hardware and software systems to accommodate its dynamic and asymmetric nature.

2.3.2 Limitations in the Image Processing Pipeline

The shadow-based positioning framework, originally designed for the needle, faced significant challenges in adapting to the forceps' geometry and functional requirements. Key issues included:

- **Keypoint Detection Accuracy:** The NN was trained on needle datasets, leading to unreliable detection performance for forceps, particularly in realistic surgical environments. The comparison of the needle and forceps is illustrated in Figure 2.18
- **ROI Management:** Maintaining an accurate ROI became increasingly difficult due to the forceps' dynamic geometry. Additionally, misdected keypoints often led to a loss of the ROI, causing the detection pipeline to fail as shown in Figure 2.19.
- **Computational Load:** The original image processing node struggled to handle the additional demands introduced by forceps-specific complexities, resulting in delays that impacted real-time operation.

These challenges underscored the need for a more robust image processing framework capable of supporting the forceps' unique demands.

2.3.3 Maintainability and Workflow Inefficiencies

The system also exhibited significant limitations in maintainability and workflow efficiency, which hindered its usability and adaptability for advanced surgical applications.

The original codebase suffered from:

- **Outdated Framework:** Built on the Rosilo framework, which lacked active support and compatibility with modern robotic platforms within the lab.
- **Hard-Coded Parameters:** The design for a 6-DoF needle necessitated manual adjustments for new instruments or tasks.
- **Minimal Modularity:** Repetitive code and a lack of helper functions made debugging

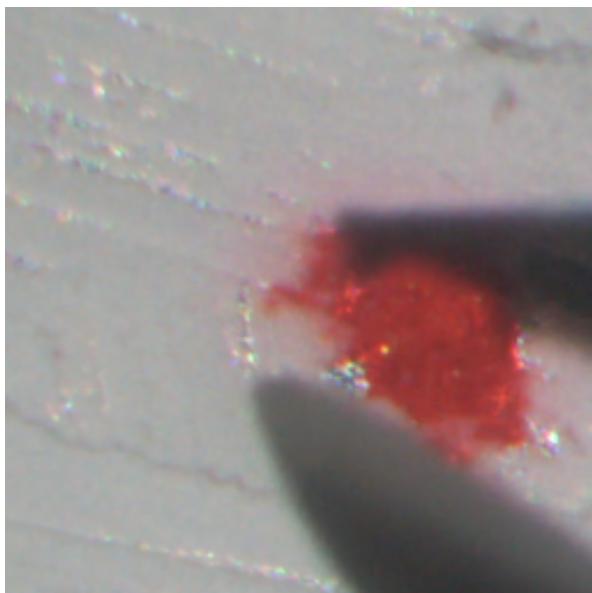
and development labor-intensive and error-prone.

- **Sparse Documentation:** Insufficient comments and documentation created barriers for new users and developers.

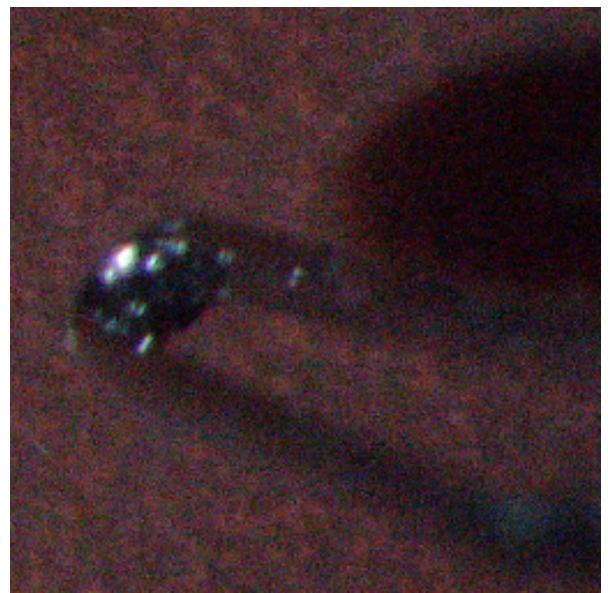
The system's workflow was cumbersome, requiring frequent manual interventions:

- **Parameter Adjustment:** Users were often required to comment and uncomment code or parameter files to modify frequently adjusted settings during robot setup and operation.
- **Calibration Complexity:** Robot calibration, essential for maintaining precision in vitreoretinal surgery, previously required switching between separate catkin workspaces and modifying parameter files manually.

These changes significantly reduced the manual workload, improved system reliability, and enhanced overall usability.



(a) The simple shape of the needle and its shadow in the autonomous positioning experiment environment.



(b) The complex shape of the forceps and its shadow in the more realistic Bionic-Eye ILM model.

Fig. 2.18: Comparing the needle and forceps in the ROI view for keypoint detection.



Fig. 2.19: Misdetection of the needle keypoints required for the ROI center point, resulting in the failure of reacquisition of the desired region.

2.4 Internal Limiting Membrane Peeling Background

2.4.1 What is ILM Peeling?

ILM peeling ^{*16} is a precise surgical procedure used in ophthalmology to treat conditions like macular holes and epiretinal membranes. The ILM, with a thickness of about $2.5 \mu\text{m}$, is the innermost layer of the retina and acts as a scaffold that can contribute to retinal traction. A surgeon removes this membrane by grasping and peeling with a specialized forceps device. The procedure helps to relieve tractional forces on the retina, allowing it to flatten and promoting healing. This can lead to improved visual acuity and a reduction in visual distortions, as the retina is able to reattach and restore its normal anatomy more effectively. ILM peeling is

^{*16}<https://youtu.be/7r1ES8YrlPk?si=DwcYkWzRviLdkI0R&t=7s>

often performed to enhance the success rates of surgeries aimed at closing macular holes and to prevent the recurrence of epiretinal membranes, ultimately improving the patient's visual outcomes.

Myopic traction maculopathy (MTM) is a condition associated with severe nearsightedness, leading to complications like retinal thickening, retinoschisis, partial macular holes, and foveal retinal detachment. Over time, MTM can progress to full-thickness macular holes and retinal detachment due to the combined traction from the vitreous cortex, epiretinal membrane, ILM, retinal vessels, and posterior staphyloma.

Surgery, specifically pars plana vitrectomy (PPV) with ILM removal, is commonly performed to alleviate this traction, often preserving or improving vision. However, this procedure can cause macular holes in 5% to 20% of cases, leading to poor visual recovery even with reoperation [37]. Figure 2.20 demonstrates the healing process from after ILM peeling to a closed macular hole.

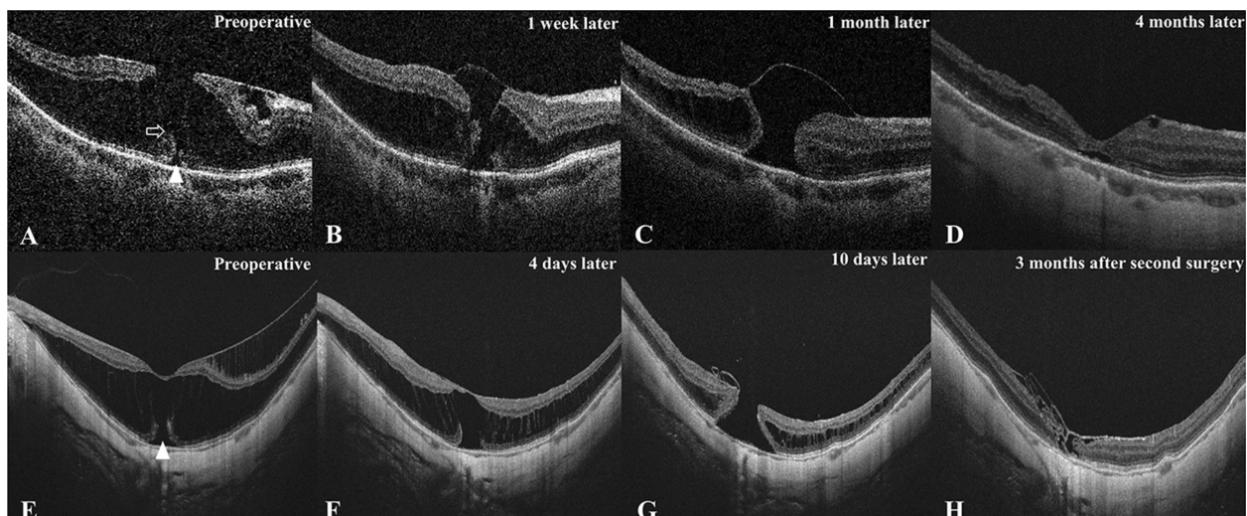


Fig. 2.20: OCT images of two patients who developed full-thickness macular hole postoperatively and closing months later [37].

2.4.2 ILM Peeling Procedure and Techniques

The general overall ILM peeling procedure [38]:

- First, the eye surgeon performs a vitrectomy. If the patient's lens needs treatment, the surgeon uses phacoemulsification to remove the lens and replace it with a single-focus artificial lens.
- During the vitrectomy, the surgeon removes the vitreous (the gel-like substance inside the eye). If present, the back boundary of the vitreous and any epimacular membrane (a thin layer on the retina) are also removed. To see the ILM more clearly, the surgeon uses a dye called indocyanine green.
- Using a microscope, the surgeon modifies a disposable needle to create a small hook. This hook gently tears the edge of the foveal ILM, a thin layer on the retina. The surgeon then carefully removes the ILM from the central part of the retina.
- For a specific technique called the inverted ILM procedure, the surgeon keeps an extra piece of ILM and flips it onto the fovea (the central part of the retina) before performing a fluid-air exchange in the eye.
- If there are any tears or weak spots in the retina, the surgeon treats them with laser photocoagulation. The surgery concludes with filling the eye with filtered air, and patients are advised to lie face down for a week to aid recovery.

A newer technique, fovea-sparing ILM peeling (FSIP), avoids peeling the ILM from the fovea itself, reducing the risk of developing full-thickness macular holes. The findings of this study suggest that the long-term anatomic and visual outcomes of PPV with FSIP for MTM are favorable and similar to those of PPV with standard ILM peeling. FSIP provides additional advantages of preventing postoperative macular holes and maintaining or improving vision.

There are two different types of FSIP techniques with and without an inverted flap. Both methods are shown in Figure 2.21. Studies show that there is no significant differences between the presence or absence of an inverted ILM flap during surgery and the results of both

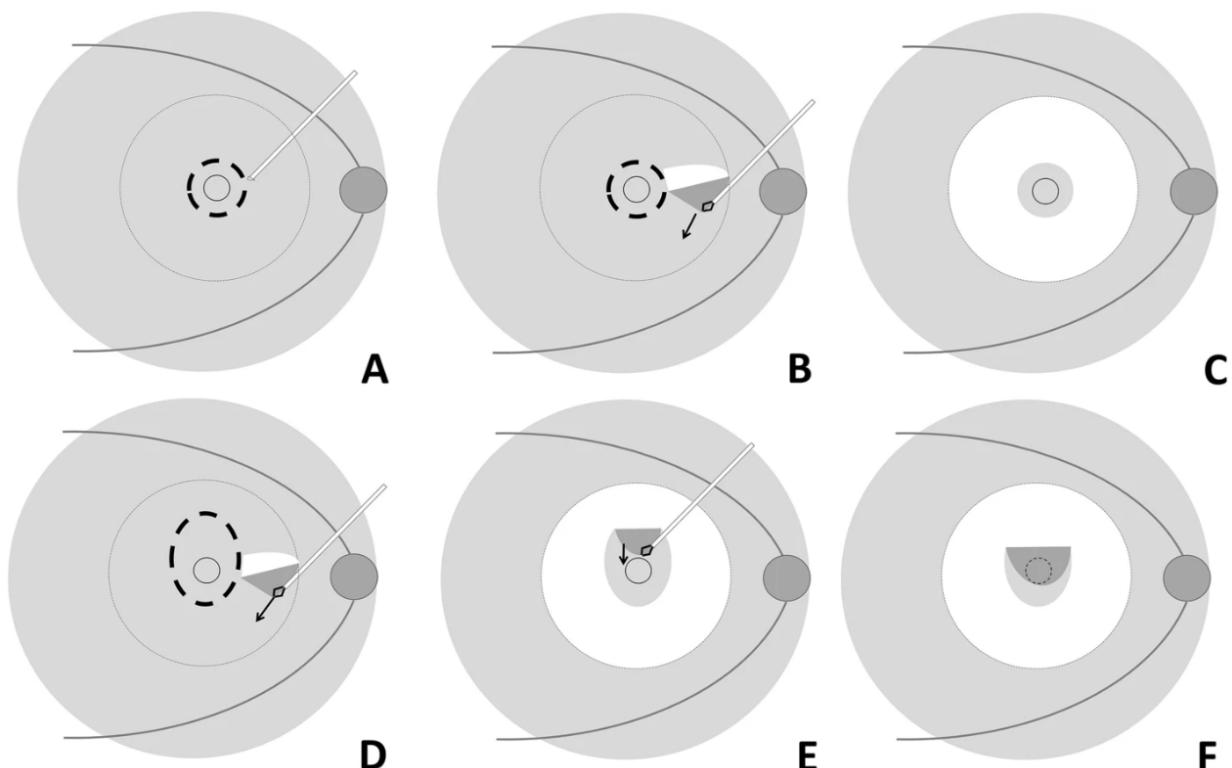


Fig. 2.21: The diagrams illustrate two methods for fovea-sparing internal limiting membrane (ILM) peeling, with and without an inverted flap [38].

methods are favorable in visual and anatomical outcomes for the treatment of MF [38].

Fovea-sparing ILM peeling procedure:

Without Inverted Flap:

- **A:** In this method, the surgeon uses a retrobulbar injection needle with a hook to carefully tear several points of ILM tissue at the edge of the reserved area, approximately one disc diameter away from the fovea.
- **B:** The peeling process continues along the boundary of the reserved area. If the peeled ILM flap deviates from this boundary, the surgeon begins peeling from a new tear site.
- **C:** The ILM is peeled off in a manner that preserves the central region of the posterior pole, indicated by a dotted line circle.

With Inverted Flap:

- **D:** An additional area of ILM, about one disc diameter above the fovea, is prepared by tearing along the reserved boundary.
- **E:** After removing the ILM around the reserved area, the extra ILM is peeled and inverted towards the foveal zone.
- **F:** The surgeon positions the prepared ILM flap onto the fovea area.

Given that microsurgery, particularly vitreoretinal surgery, is highly challenging even for the most skilled surgeons, there is a growing demand for robotic systems to enhance operational accuracy and ensure greater patient safety.

Chapter 3 Advancing the Robotic Capabilities with Forceps Integration

Building on the system limitations identified in Chapter 2, this chapter describes the modifications required to integrate a forceps device into the Vitreoretinal SmartArm system. These updates include hardware changes to accommodate the additional degrees of freedom required by the forceps and software enhancements to adapt the kinematic framework and improve system maintainability.

3.1 Integration of the Forceps System

The integration of the forceps device introduces two additional degrees of freedom: rotation and grasping. The rotational capability allows the forceps to rotate independently of the robot's arm, enabling precise orientation without requiring the entire robotic system to move. The grasping function facilitates securely holding the ILM for peeling, a critical aspect of vitreoretinal surgery.

This section outlines the design and components of the forceps system, including:

- The forceps instrument, designed to perform the ILM peeling task with interchangeability in mind.

- The forceps attachment, equipped with brushed DC motors for rotation and grasping.
- A control box integrated with encoders to manage motor control and provide feedback for precise operation.
- PCI boards installed on the system's computer to handle input/output operations and ensure seamless communication between hardware and software components.

Figure 3.1 illustrates the hardware architecture of the Vitreoretinal SmartArm system, highlighting the integration of the forceps device and its supporting components.

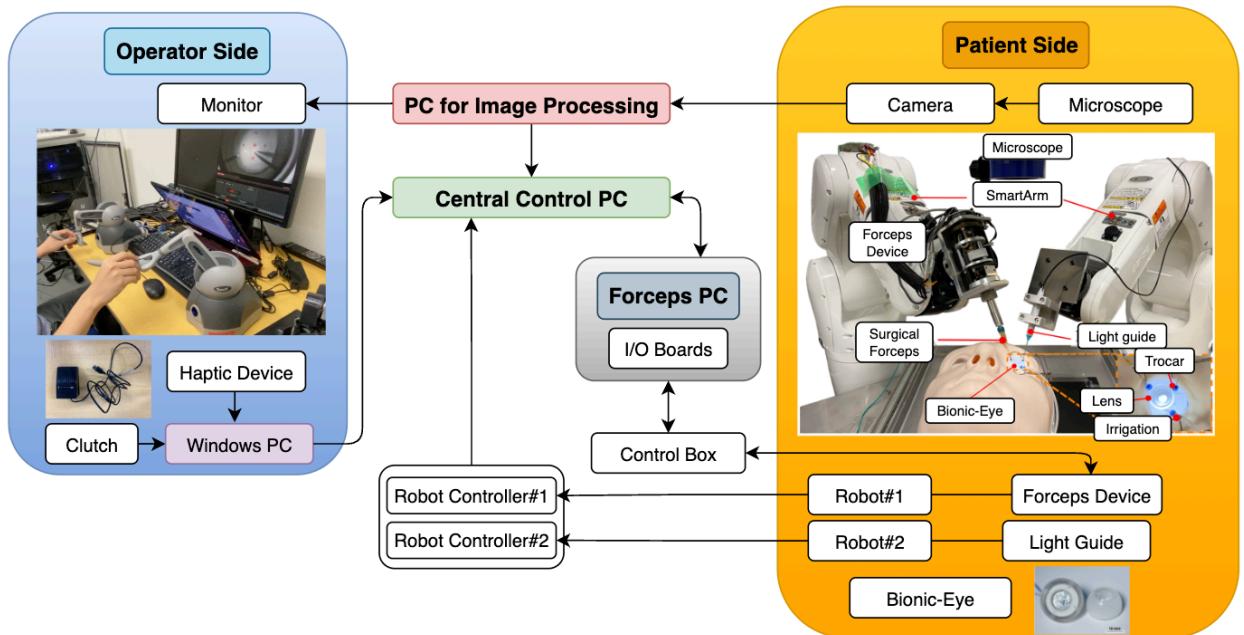


Fig. 3.1: The hardware setup of the Vitreoretinal SmartArm system with integrated forceps device.

Previous research within the laboratory utilized forceps devices with the same robotic arms (DENSO VS050, DENSWAVE, Japan), laying a robust foundation for this study. The existing hardware infrastructure, as shown in Figure 3.1, ensures compatibility and reliability, allowing for streamlined integration and reduced development time [28].

Similarly, the software architecture, depicted in Figure 3.2, incorporates the key elements required to operate the system, including ROS nodes for communication, the image processing pipeline, and robot control algorithms. Building upon this established software frame-

work, the current study introduces enhancements to address the specific requirements of ILM peeling, ensuring precise control and improved functionality of the forceps device.

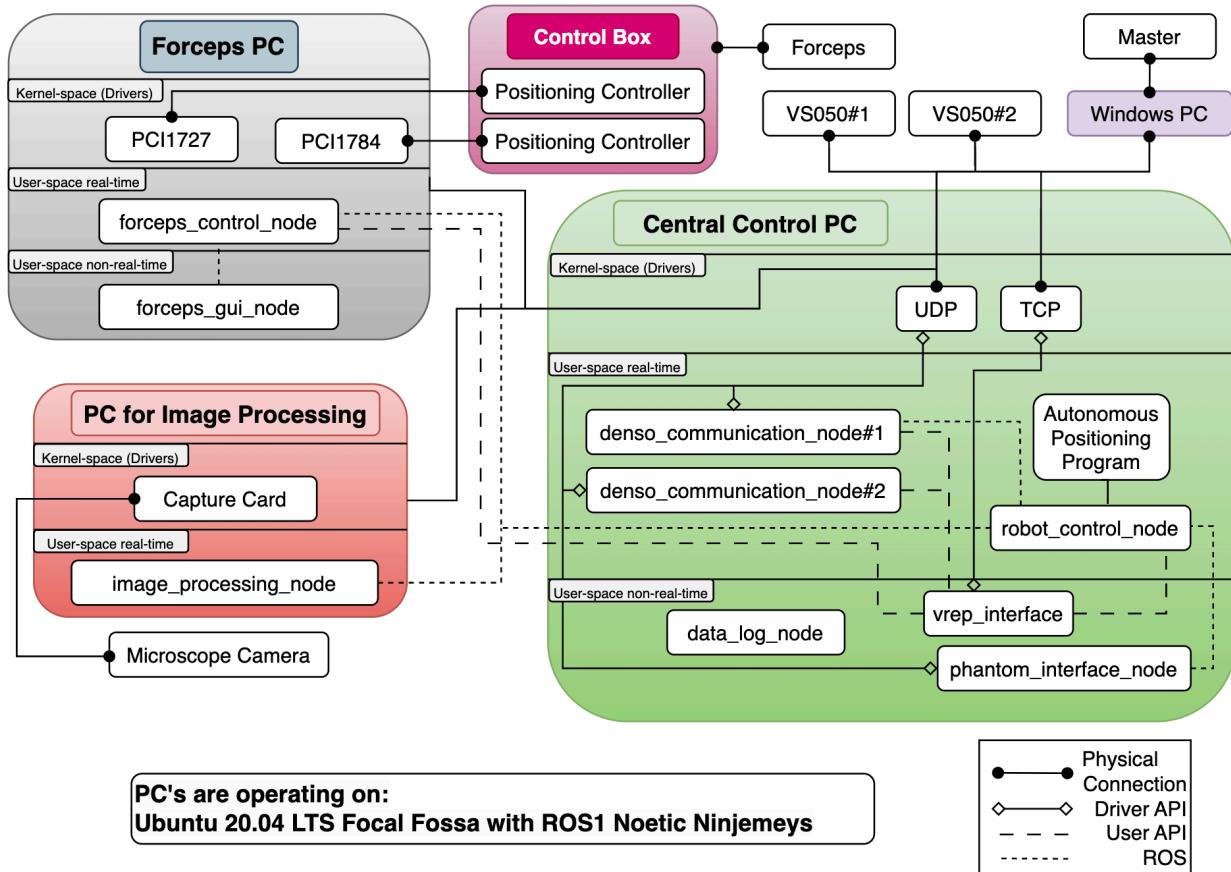


Fig. 3.2: The software setup of the Vitreoretinal SmartArm system with integrated forceps device.

3.1.1 The Forceps Primary Surgical Instrument

The Vitreoretinal SmartArm system for ILM peeling surgery requires a certified forceps tip that is easily interchangeable and attachable. The Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga^{*1}, Alcon, USA), as shown in Figure 3.3, is an ideal choice due to its specialized design, rigidity, compact form factor, and quick-lock connector. Originally designed

^{*1}<https://alcon.widen.net/s/zrlg779y10>

to accommodate different grips based on a surgeon's preference, this forceps instrument is familiar to the surgeon along with being well-suited for the robotic attachment. Additionally, it is readily available as an off-the-shelf product, making it a practical and accessible option for integration. The dimensions^{*2} can be found via Table 3.1 with a corresponding image via Figure 3.4.

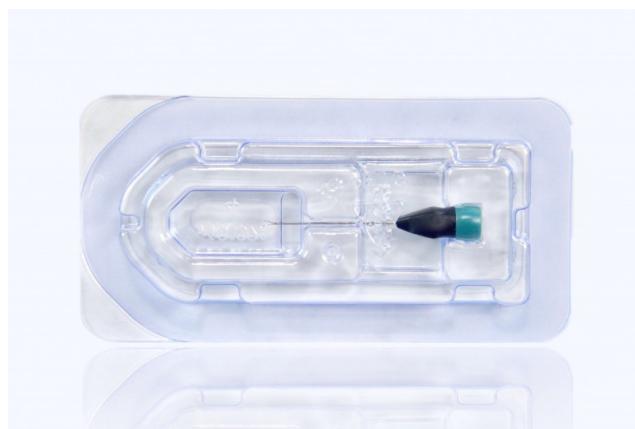


Fig. 3.3: The primary surgical instrument used in this thesis: Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga.

Gauge	Holding force	Overall Length	Length of tip	Length of platform	Width of platform
25 Ga	25g	27mm	3.4mm	0.12mm	0.27mm

Table 3.1: Specifications of the the Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga.

The opening and closing mechanism of the forceps instrument is controlled by a cylinder that extends into its base. When the cylinder is pushed, the forceps closes; when released, it opens.

3.1.2 Forceps Robot Attachment

The main robot attachment, as shown in Figure 3.5 is a custom-designed device mounted to the end effector of the primary surgical instrument arm. It consists of two key compo-

^{*2}<https://bikomed.bg/wp-content/uploads/Broshures/GRIESHABER-Product-Catalog.pdf>

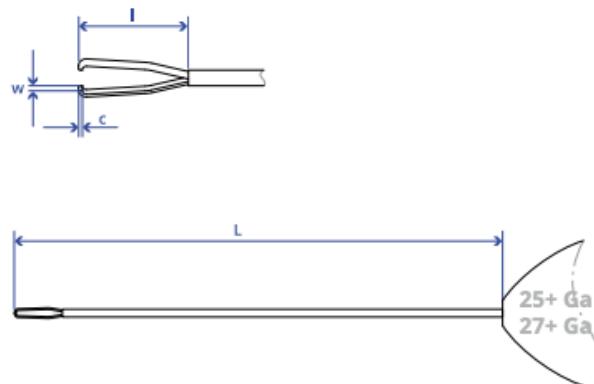


Fig. 3.4: Dimension diagram of the the Alcon® Grieshaber® Advanced DSP Tip ILM Forceps 25Ga.

nents: the main housing and the rotational housing. Two brushed DC motors (gear, motor, and sensor combinations, Maxon Group^{*3}, Switzerland) are incorporated, each equipped with specified gear sets and sensors to facilitate the rotation and grasping functionalities of the forceps. Detailed specifications for both motor combinations are provided in Table 3.2.

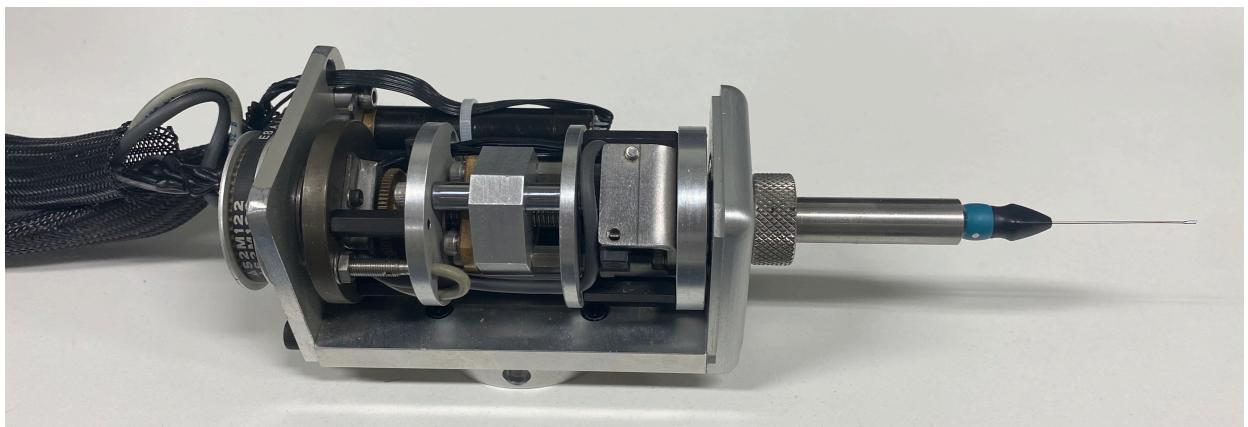


Fig. 3.5: The forceps robot attachment with its rotational and grasping capabilities.

The rotational motor, mounted on the main housing, drives the rotational housing via a gear and belt system. The rotational housing contains most of the attachment's components, including a sliding base and its mechanism, two limit switches, the forceps motor, and the

^{*3}<https://www.maxongroup.com/en>

Purpose	Gear	Motor	Sensor
rotation	$\varnothing 13\text{mm}$, 0.02 - 0.35Nm	$\varnothing 13\text{mm}$	2-ch, 16 CPT
grasping	$\varnothing 10\text{mm}$, 0.01 - 0.15Nm	$\varnothing 10\text{mm}$	2-ch, 12 CPT

Table 3.2: Specifications for the rotation and grasping Maxon Group motorized gear sets.

forceps instrument itself. The limit switches sensors are critical for determining if the system is moving and the homing position for both the rotation and the maximum grasping range of the forceps. This enables the system to calibrate and establish relative positions based on the "zero" reference.

The forceps motor, responsible for controlling the grasping mechanism, drives a sliding component (rotational to translational) that pushes a cylinder into the base of the forceps instrument to close it. The grasping limit switch is mounted on the sliding base to provide accurate positioning for the grasping operation.

The motor encoder, part of the motor combination, is a sensor that provides real-time feedback on the motor's position, speed, and direction of rotation. It generates signals that indicate how far and in which direction a motor has moved, enabling the system to monitor and control its movement accurately. Encoders are critical in ensuring fine-grained motor control, as they provide the data needed by a positioning controller to make real-time adjustments for precise and reliable operation.

The motors and sensors are pinned to a 24 I/O connector, which is wired to the control box. Precaution is necessary for the wiring as most of the wires twist along with the rotational housing.

3.1.3 Control Box

The control box allows the communication between the feedback from the forceps device to the desired positioning provided by the user/program. The control box, as shown in Figure

3.6, contains two positioning controllers^{*4} (Maxon Group, Switzerland) that control both of the motors and an AC 100-240VAC power supply^{*5} (Mean Well^{*6}, Taiwan). The wiring diagram can be viewed through Figure 3.7.



(a) Front - rotation and forceps switches, 24-pin connector to the forceps attachment.

(b) Back - power switch, 2x 36-pin connectors to the computer's encoder and I/O board.

Fig. 3.6: The forceps control box containing the two motor positioning controllers.

The digital positioning controllers manage the precise movement of the motors by processing feedback data, provided by the encoders on the motors. A controller compares the desired position, speed, or motion trajectory to the actual motor output and adjusts the motor's input accordingly to minimize errors. This closed-loop system ensures accurate positioning, smooth motion, and repeatability, making it essential in precise control of the surgical robotic device.

The motor combinations are configured and tuned using Maxon's EPOS2 software^{*7}. The setup wizards provide a user-friendly interface for configuring inputs and outputs, setting up the motor combination, and fine-tuning the motors integrated into the mechanical system. Additionally, the software allows for manual motor control, a critical feature for system de-

^{*4}<https://www.maxongroup.com/maxon/view/product/control/Positionierung/390438>

^{*5}<https://www.radwell.eu/de/buy/mean-well-mdr-10-24/4902754.html>

^{*6}<https://www.meanwell.com/>

^{*7}<https://maxonjapan.com/wp-content/uploads/manual/epos/EPOS2-Application-Notes-Collection-En.pdf>

bugging and troubleshooting.

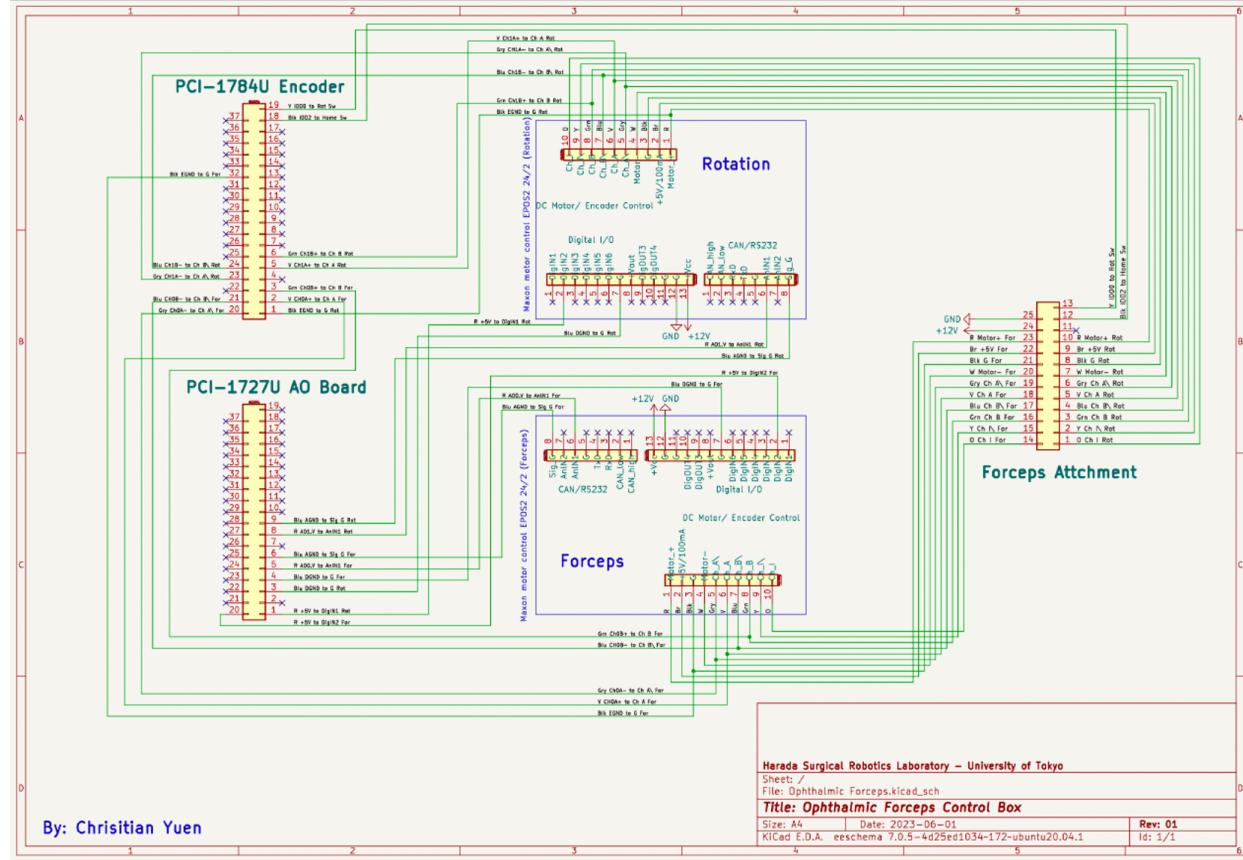


Fig. 3.7: Circuit diagram of the control box connected to the forceps attachment and I/O boards.

3.1.4 Forceps Computer

The current system includes a dedicated computer for the forceps device, equipped with an I/O board (PCI-1727U^{*8}) and an encoder board (PCI-1784^{*9}, Advantech, Taiwan), both featuring 36-pin connectors. The I/O board facilitates communication between the computer and the control box, while the encoder board, though seemingly redundant, plays a crucial role in the system.

*⁸https://www.advantech.com/en/products/1-2mlk8x/pci-1727u/mod_4c575a4f-2926-4c5e-8770-54aa219fcc5d
*⁹https://www.advantech.com/en/products/1-2mlk9r/pci-1784u/mod_96795d5f-1e79-4f65-94e6-60dd07ec8133

The computer-installed encoder is intentionally designed to provide redundancy, enhancing the safety and reliability of the system by ensuring continuous operation and accurate feedback. It also enables cross-checking with the motor-mounted encoder to verify the accuracy of measurements. Even though the motor-mounted encoder is highly accurate with instant feedback, a key advantage of the computer-installed encoder lies in its ability to monitor the mechanical transformations occurring between the gears and belt, which the motor-attached encoder cannot capture, as it only measures shaft rotation.

This additional encoder allows the system to control limit switches, ensuring that the forceps can be accurately monitored at its home position or during movement. Furthermore, the computer-installed encoder supports system-wide calibration of the forceps device, enabling more precise and consistent operation.

As a side note, a dedicated computer for the forceps device is unnecessary and integration to the central computer is preferred. Unfortunately, the PCI cards are not compatible with newer computers that have PCI-E slots. Ideally, both PCI-E cards are to be installed on the computer that handles the kinematics of the robots for faster communication.

3.1.5 Escon Controller

Communication between the main computer, which processes kinematics, and the forceps device, along with its robotic manipulator, is managed through the Escon Controller. This controller allows the manipulator and forceps device to function seamlessly as a unified robotic system.

To configure motor settings and establish connections between the encoder and motor controllers, the Escon Controller Manager is used. This tool enables the configuration of parameters such as the PID controller, channels, motor specifications, joint limits, and homed sensor states. Additionally, the Escon operation check node facilitates real-time monitoring and manipulation of individual motors and their states, as illustrated in Figure 3.8.

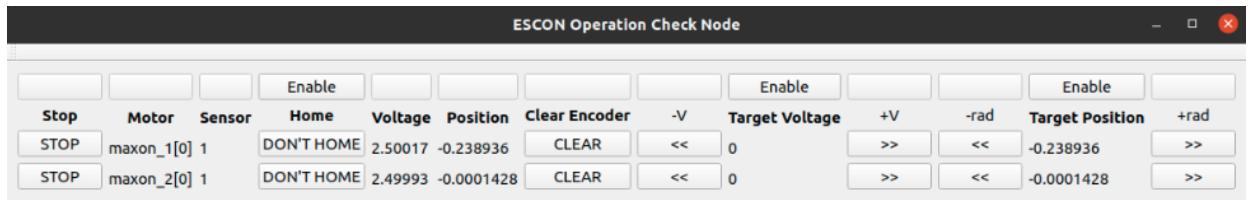


Fig. 3.8: The Escon operation check node with motor control and state information.

3.2 Progressing the Robot End Effector and Its Kinematic Design

Integrating the rotational 7th joint of the forceps device necessitates several updates: modifications to the robot kinematics and its parameters, enhancements to the Coppelia simulator scene, and enabling operator control for opening and closing the forceps.

3.2.1 The Seventh Joint

The kinematic parametrization of the main surgical instrument robot was updated to incorporate the seventh joint. Detailed information on the Denavit-Hartenberg (DH) parameters and DENSO Corporation's proprietary parameters can be found in Section 8.2.3. The DENSO parameters were chosen for the Vitreoretinal SmartArm system due to their superior accuracy in calibrating the robot manipulator.

The DQ Robotics package provides a variety of robot models for simulation and control. Initially, the forceps device was conceptualized as a separate robotic entity attached to the main robot manipulator, utilizing a whole-body robot modeling approach. However, the whole-body functionality in DQ Robotics is limited and less rigorously tested compared to widely used models like the serial manipulator DENSO model.

Consequently, the Vitreoretinal SmartArm system continues to employ the serial manipula-

tor DENSO model, augmented with modified DENSO proprietary parameters for the forceps. This ensures accurate and reliable kinematic parametrization.

3.2.2 Adding Rotation to the Constrained Optimization Algorithm

The Vitreoretinal SmartArm system employed a constrained optimization algorithm on the follower side that focused solely on the translational movements of the end-effector. This approach is sufficient for instruments like the surgical needle and light guide, which are symmetric along its length and do not require consideration of end-effector rotation. However, the forceps device introduces new requirements, necessitating the integration of the rotational kinematic structure of the end-effector.

To enable rotation, the system was integrated using the same constrained optimization approach described in [39] and detailed in Section 8.3.

Parameters

The parameters for the constrained optimization algorithm are configured to prioritize the 7th rotational joint for rotation. One of the parameters allows the seventh joint to enable or disable the rotational kinematic. The translation and rotation weight, α , is set to 0.999995, heavily favoring rotational gains. The robot prioritization weight, β , is assigned a value of 0.99, giving significant priority to the primary surgical instrument. Additionally, the joint weight matrix, Λ , is designed to emphasize the 7th joint, ensuring its rotational movement is prioritized over the other joints in the robot.

Kinematics Bug

A bug has been identified within the implementation of rotational kinematics in conjunction with the orbital manipulation framework. This issue persists regardless of whether orbital manipulation is enabled or not. While the impact is minimal when orbital manipulation is disabled, the bug causes the eyeball to gradually rotate toward one side without returning to its original position, as illustrated in Figure 3.9. To mitigate this issue during experiments, the

remote center of motion (RCM) points should be reinserted and reset periodically to maintain stability and accuracy.

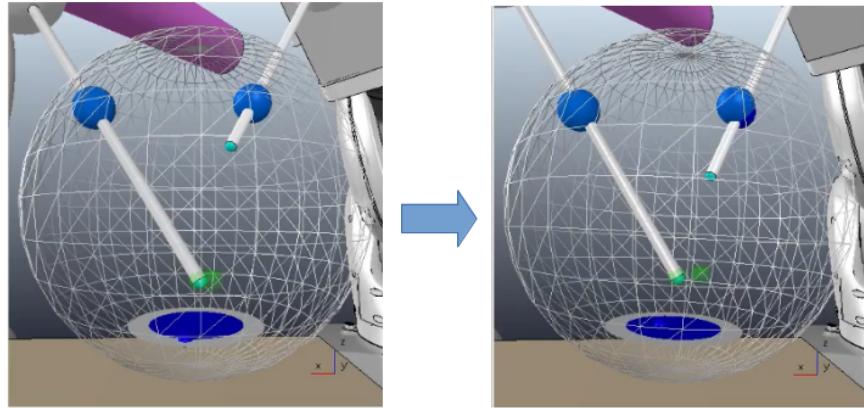


Fig. 3.9: Bug involving the implementation of rotation to the forceps with the Orbital Manipulation kinematics framework. The eyeball slightly shifts to a side and does not return to center.

3.2.3 Simulator Integration

The simulator required integration and maintenance to support the additional joint for the forceps device. To verify the proper implementation of kinematics, a simple simulation scene was created featuring the robot manipulator and the forceps device. A seventh joint, representing the rotational movement of the forceps, was added as shown in Figure 3.10. Basic joint commands and specific poses were tested to validate the kinematic framework. Once verified, the updated primary surgical instrument robot was incorporated into the more complex vitreoretinal surgery simulation scene.

The vitreoretinal scene also required updates and refinements. The existing robot manipulator models were disjointed, necessitating the import of new models for accurate representation. Additional views of the eye were integrated to enhance the user's understanding of activities within the workspace. Furthermore, the workspace and its hierarchical structures were reorganized for more intuitive navigation and improved usability. The Coppelia simulator scene of the vitreoretinal workspace is illustrated in Figure 3.11

3.2.4 Teleoperating the Forceps

The haptic devices utilized for teleoperation serve as the primary interface for inputting translational and rotational commands. These rotational inputs have been seamlessly integrated into the system's kinematic framework, enabling precise manipulation. Additionally, the buttons on the haptic device have been configured to control the opening and closing of the forceps, using a single-button. When pressed upon, the forceps closes and reopens upon release. This method is implemented by varying the voltage sent to the motor, contingent on the state of the button being pressed. The operation of the forceps inside the Bionic-Eye ILM model are illustrated in Figure 3.12.

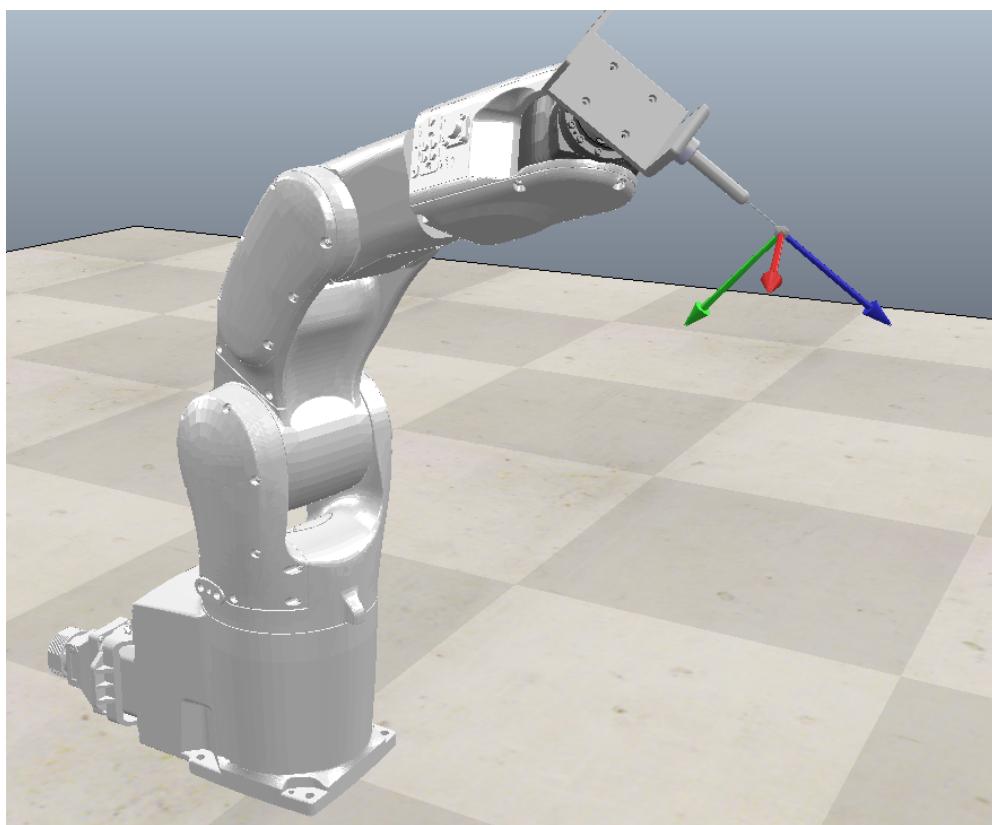


Fig. 3.10: A simple simulation scene of the robot manipulator with forceps device. The different axis of the end effector are shown.

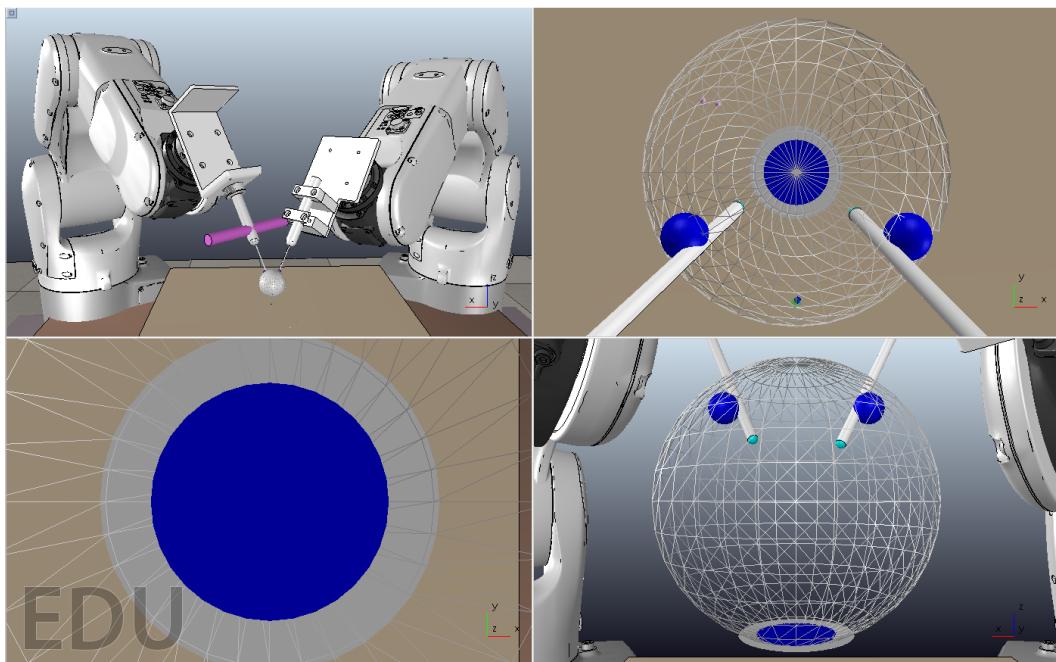


Fig. 3.11: Vitreoretinal workspace scene in Coppelia simulator with both tools inserted at the RCM points. The RCM points are set in this position before translating to the initial positions.



Fig. 3.12: The forceps grasping the ILM inside the Bionic-Eye ILM model.

3.3 System Framework Enhancements and Code Optimization

Extensive enhancements to the framework and workflow, including transitions and refactoring, were undertaken to enhance the system's support, maintainability, and functionality.

3.3.1 Streamlining System Architecture and Workflow Optimization

The robotic system was initially developed on a framework that is no longer supported by its developer, resulting in limited expertise available in the laboratory for its maintenance. Transitioning to the SmartArmStack ROS1 framework was essential due to its modern structure and syntax. This transition provides enhanced support, access to advanced features, and alignment with the streamlined workflows used across other robotic systems in the laboratory.

Previously, users were often required to manually comment and uncomment code or parameter files to adjust frequently modified settings during robot setup and usage. To address this, the JSON DH parameter working directories are now integrated into the bash file, ensuring consistency across all programs and workspaces on the central computer system. Additionally, specific sets of YAML configuration and launch files have been created to work cohesively, enabling tailored program execution without the need for direct file modifications.

Robot calibration, which is critical due to the sensitivity and precision required for vitreoretinal surgery, has also been significantly streamlined. Calibration no longer requires a separate catkin workspace, reducing the need for frequent workspace changes. The process has been optimized to allow both robots to be calibrated directly within the program, eliminating the need to quit and modify parameter files manually. The calibration user interface, shown in Figure 3.13, facilitates this improved workflow.

Each robot's calibration data is now saved to its respective file, with the program reading the original factory JSON calibration files and outputting calibrated JSON files separately. This approach preserves the original files for easy interchangeability and prevents accidental modifications, further enhancing reliability and usability.

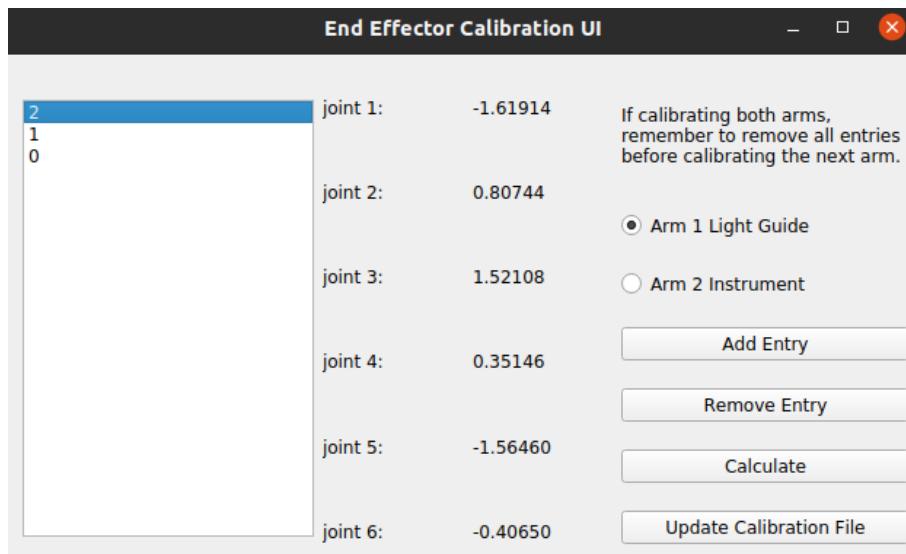


Fig. 3.13: Updated End Effector Calibration UI with robot selection and calibration file update button

3.3.2 Code Refactoring

Code refactoring plays a critical role in addressing long-standing software challenges and by preparing the system for future enhancements. A key objective is to improve maintainability, streamline development, and enhance flexibility, particularly in accommodating the robotic system's evolving requirements.

One of the primary issues resolved was the hard-coding of six degrees of freedom throughout the workspace, which limited scalability and complicated updates. A dynamic variable was introduced to determine the number of joints in the robot, enabling the seamless integration of the seventh joint and providing the flexibility to support additional joints in the future.

This improvement makes navigating the extensive workspace more efficient and simplifies potential expansions. The data loggers during operation for both teleoperation and automation were also updated to accommodate the additional joint.

Additionally, a helper function class was developed to handle repetitive calculations required for positioning the robots in teleoperation and automated processes. This class consolidates essential operations, including the computation of poses, Jacobians, and error metrics used to determine decision variables for quadratic programming. The class also updates the optimization functions to guide robots to desired positions, enhancing code organization and clarity. These changes are particularly valuable for the complex multi-step automation processes discussed in the subsequent chapter.

Finally, detailed comments were added throughout the codebase to make it more accessible to new users, facilitating quicker understanding and adoption of the system. By addressing these challenges, the refactoring efforts have transformed the code into a more maintainable and extensible foundation, ensuring it can effectively support future developments and improvements.

Chapter 4 Shadow-Based Positioning Framework

Achieving autonomous vitreoretinal surgery depends on precise image detection to identify and process key areas within the surgical workspace. Critical to this process are the tip of the surgical instrument and its shadow, which are fundamental for accurate positioning. This chapter addresses the challenges of adapting the Vitreoretinal SmartArm system's shadow-based positioning framework for the forceps device, whose dynamic geometry and functional variability significantly increase complexity.

To meet these challenges, a dual-neural network (NN) approach was implemented, separating keypoint detection and region of interest (ROI) management. A specialized ROI detection NN was introduced to improve stability and accuracy, including shaft keypoint detection. The chapter details the development of these models and their integration into the updated shadow-based positioning framework, with a focus on improving image processing and algorithmic precision. Finally, an experiment is outlined which will evaluate the feasibility and performance of these updates, offering insights into their potential to advance robotic vitreoretinal surgery.

4.1 Rationale for Splitting Keypoint Detection

Integrating the forceps device into the Vitreoretinal SmartArm system introduced significant challenges for keypoint detection. Unlike the surgical needle, the forceps' geometry is dynamic, varying with both orientation and degree of closure. This complexity led to inconsistencies in detecting keypoints, particularly the shaft and tip, within the constrained ROI.

Initially, a single NN was used to handle both keypoint detection and ROI management. However, this approach proved inadequate due to the following challenges:

- **ROI Stability:** The original method calculated the ROI center as the midpoint between the instrument tip and its shadow with the previous frame. However, misdetections, especially with the complex geometry of the forceps, often caused the desired ROI region to be lost without recovery. Additionally, the instruments needed to start within the ROI region, increasing the demand for flexible, reliable, and robust ROI management.
- **Shaft Keypoint Detection:** Shaft detection, originally part of the main keypoint detection NN, proved unreliable when constrained to the small ROI, especially due to the hole created by the forceps arms. By moving shaft keypoint detection to the ROI NN, which observes the entire microscope workspace, detection accuracy significantly improved.

To address these limitations, the detection framework was divided into two specialized NNs:

1. **Main Keypoint Detection NN:** Dedicated to identifying the tip and shadow of the forceps within the ROI.
2. **ROI Detection NN:** Responsible for determining the center point and dimensions of the ROI and for handling shaft keypoint detection across the entire workspace.

This division enabled independent optimization of each NN, improving overall system performance. By allowing the ROI NN to analyze the full microscope workspace, the system became more adaptable to variations in instrument position and geometry, ensuring enhanced

stability and precision in challenging scenarios.

4.2 Redesigning the Main Keypoint Detection for Forceps Adaption

Adapting the Vitreoretinal SmartArm system for forceps required a specialized NN to manage the tool's complex geometry and dynamic behavior. Unlike the surgical needle, the forceps' shape varies with orientation and degree of closure, complicating accurate detection. These challenges were further amplified by environmental variability in the Bionic-Eye ILM model, including glare, dim lighting, and textured backgrounds.

A major limitation was the unreliability of shaft keypoint detection within the constrained ROI, particularly in certain poses. To address this, the system incorporated preimage processing, postprocessing adjustments, and a dedicated ROI detection NN. These enhancements improved the system's ability to detect and position the forceps accurately. The relocation of shaft keypoint detection to the ROI NN is further discussed in Section 4.3.

4.2.1 Adapting the Neural Network

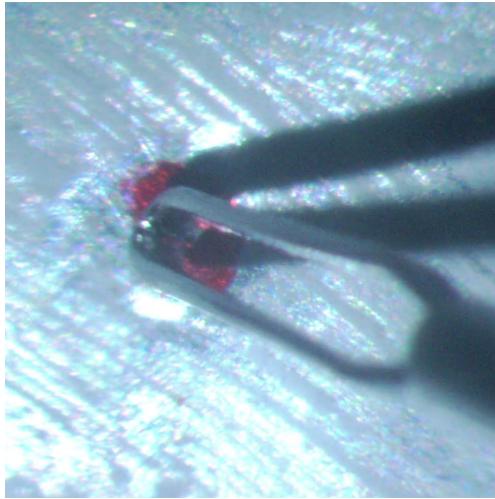
Significant updates to the keypoint detection framework were required to accommodate the forceps' complex geometry and challenges, especially in the Bionic-Eye ILM model. The described challenges were mitigated through pre- and postprocessing enhancements.

To overcome these challenges, the following enhancements were implemented:

- **Preimage Processing:** Techniques to improve image clarity and reduce noise.
- **Postprocessing Adjustments:** Methods to stabilize and refine detected keypoints.

Figure 4.1 illustrates examples of challenging orientations for keypoint detection. These updates were designed to enhance overall system performance, ensuring greater adaptability

and robustness. Details on the implementation and effectiveness of the ROI detection NN are provided in Section 4.3.



(a) Forceps occluding its shadow in autonomous positioning environment.



(b) Shadow between forceps arms in Bionic-Eye ILM model.

Fig. 4.1: Different orientations the updated NN must recognize.

4.2.2 Adjusting the ROI Size

To accommodate the increased detail in a larger image, the starting ROI size was expanded from the previously used 256 x 256 pixels to 512 x 512 pixels. This larger ROI provides the NN with more information, prioritizing overall accuracy over precision by offering a broader view of the area of interest. Initially, the concept was to use the full 512 x 512 ROI for detection. However, in live system applications, this approach presents two primary drawbacks:

- **Increased Computational Load:** The convolutional layers must process four times as many pixels, significantly slowing detection speed.
- **Larger Message Size:** The ROS system must transmit a message four times larger, further contributing to processing delays.

To address this issue, the ROI image is downscaled to 256 x 256 pixels. This adjustment

retains the essential area of interest ensures smoother real-time performance without while preserving most of the critical information for keypoint detection.

4.2.3 Pre-Image Processing for Enhanced Keypoint Detection

Keypoint detection for the forceps device presents unique challenges due to its complex geometry and the low-light environment of the Bionic-Eye ILM model. To overcome these difficulties, a sequence of pre-image processing techniques is applied to the resized and cropped ROI image. These steps are designed to enhance clarity, reduce noise, and emphasize critical features, ensuring more reliable performance of the keypoint detection NN.

The first step involves bilateral filtering, a noise-reduction technique that preserves edges by combining spatial (domain) and intensity (range) Gaussian weighting. Unlike Gaussian blur, which can overly smooth edges, bilateral filtering maintains the integrity of edges and boundaries while effectively reducing noise. This ensures that important structural features of the image remain sharp and well-defined.

Next, the image undergoes contrast-limited adaptive histogram equalization (CLAHE) to locally enhance contrast and improve the visibility of features in low-contrast regions. CLAHE divides the image into small regions, adjusts the intensity distribution within each region, and then merges them to create an evenly enhanced image. A clip limit of 2.5 is used to control contrast amplification and prevent artifacts or noise. The image is divided into a grid of 6x6 tiles to balance local contrast enhancement across the image.

Following contrast adjustment, min-max normalization is applied to stretch pixel intensities across the full range of 0-255. This standardization improves overall contrast and ensures consistency in pixel intensity distribution, which is critical for subsequent processing steps like edge detection.

The final step involves Sobel edge detection, which calculates image gradients in both the

x and y directions to emphasize edges. These gradients are merged with the original image using a weighted ratio of 0.7:0.3 (original image to edge image). This blending ensures that the resulting image highlights critical boundaries while retaining sufficient context from the original image.

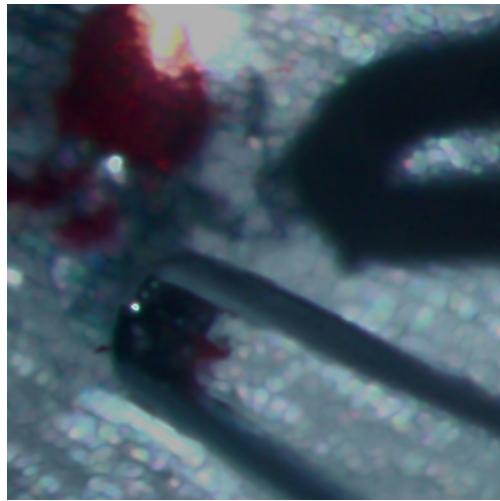
By applying this sequence of pre-image processing techniques, the ROI image is optimized for keypoint detection. The resulting image provides the main NN with clearer and more distinguishable features, allowing it to perform more reliably in noisy or low-contrast environments.

4.2.4 Dataset and Augmentation

The NN must also be trained on a sufficient dataset size with variability to accommodate the plethora of orientations and situations. Images from both the autonomous positioning and Bionic-Eye model, about half and half, are provided to accommodate both environments. As a result a dataset of 1019 images were labeled and prepared for the NN. A grand total of 6000 labeled images after image augmentation were used for training the keypoint detection NN. Some of the image augmentations include minor rotation, various blurs like Gaussian blur, increased noise, and random brightness and contrast. A few examples of the augmented images are shown in Figure 4.2.

4.2.5 NN Training Results

The neural network was implemented using the PyTorch library [40] with a ResNet-18 encoder pretrained on ImageNet. Mean squared error (MSE) as the loss function, a weight decay of 1×10^{-5} , and a softmax activation were employed. The model was trained for 80 epochs on an NVIDIA GeForce RTX 3070 GPU using the Adam optimizer, starting with a learning rate of 1×10^{-3} and subsequently reducing it to 1×10^{-4} and then 1×10^{-5} . A batch



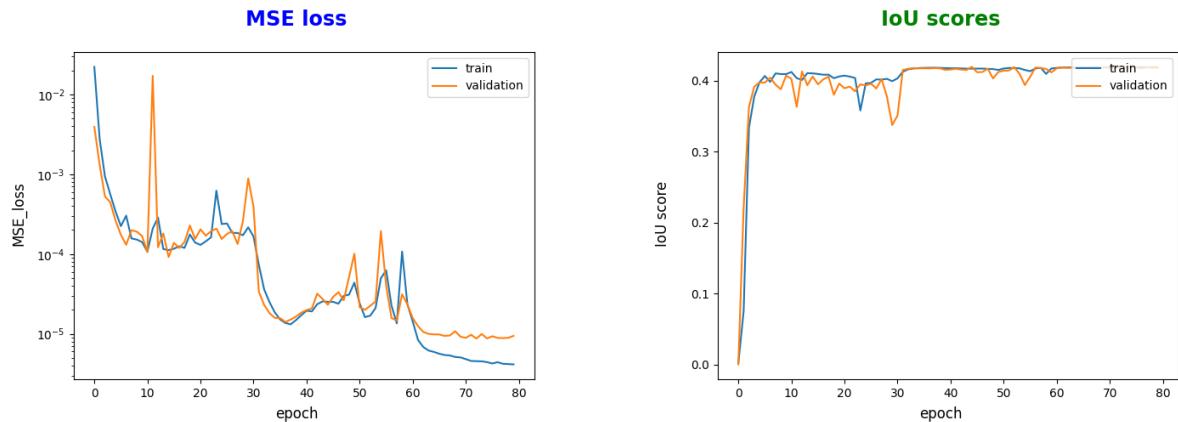
(a) Autonomous positioning environment



(b) Bionic-Eye ILM model

Fig. 4.2: Preprocessed and augmented images for keypoint detection NN training.

size of 32 and 8 worker threads for data loading was used. On the test set, the model achieved an MSE loss of 1×10^{-5} and an IoU score of 0.4 as illustrated in Figure 4.3.

(a) Learning Curve - training and validation MSE dropped below 1×10^{-5} .

(b) IoU Scores - training and validation scores hover around 0.4 for most of training.

Fig. 4.3: Main Keypoint Detection Model Training and Validation Results - The model trained stably and does not drastically overfit.

4.2.6 Enhancing Keypoint Detection Stability and Noise Mitigation

To improve the accuracy and stability of keypoint detection for the forceps tip and its shadow, techniques such as moving average smoothing and outlier detection are employed. These enhancements aim to mitigate noise and ensure reliable performance during real-time operation.

An exponential moving average (EMA) is applied to the detected keypoints, prioritizing recent detections while considering historical data. This approach is particularly well-suited for scenarios where the detected keypoints are accurate but not highly precise. By averaging results, the EMA helps align the detected keypoint closer to its true pixel location, effectively reducing the impact of minor noise in the system. Since the instrument is in constant motion, emphasizing the latest detected location ensures that the keypoint detection does not lag behind the instrument's movement. The implementation uses a deque with a customizable length to store previous points, weighting them exponentially. A fast NN is essential for this method to prevent lag in output smoothing due to previous data.

Outlier detection is also critical to address occasional false keypoint detections that deviate significantly from the desired location. To manage this, a user-defined distance threshold is introduced. If the difference between the previous keypoint and the newly detected keypoint exceeds this threshold, the system rejects the new point and instead retains the previous one in the deque. This prevents erratic keypoint shifts caused by false detections.

Optimizing the threshold value is crucial for effective outlier detection. A threshold that is too large may fail to filter out false detections, while a threshold that is too small could incorrectly flag valid detections as outliers. Another potential issue is cascading errors, where the keypoint shifts to false detections and struggles to return to the correct pixel as illustrated in Figure 4.4. Proper calibration and testing of the threshold value mitigate these risks to

ensure robust performance.

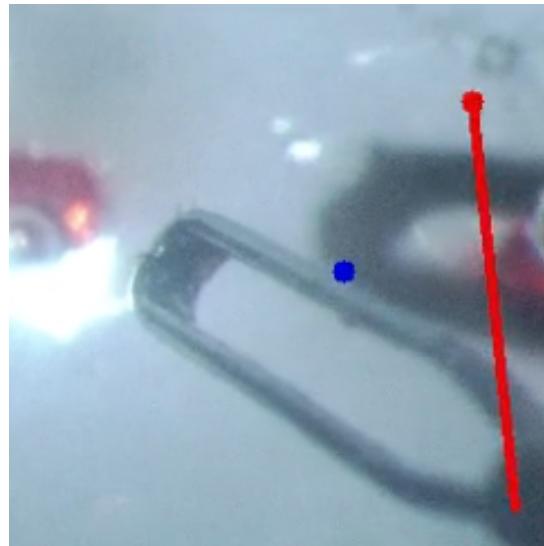


Fig. 4.4: Example of severe keypoint detection error of the forceps tip.

4.3 Adaptive ROI and Shaft Point Keypoint Detection

Establishing the ROI center point using previously detected keypoints may seem practical, but it has a critical flaw: it relies heavily on highly accurate detection with minimal outliers. A single incorrect pixel detection can shift the ROI in the wrong direction, potentially excluding the forceps, shadow, or shaft from the ROI. A cascading effect can occur, causing the ROI to drift from the desired region, breaking the program. While this approach worked quite well for the robust detection of the surgical needle, it is unsuitable for the more complex geometry of the forceps. Thus, a new method for determining the ROI center point became necessary. Additionally, the instrument shaft keypoint is also migrated to this NN for more accurate detection.

4.3.1 Lightweight Methodology and Neural Network Design

Since the primary purpose of the ROI is to reduce the image size processing for keypoint detection, an efficient and lightweight method is required. While adding an additional NN may seem resource-intensive, careful optimization can minimize its computational burden. The ROI NN leverages the same base image dataset, but with the whole workspace view, as the keypoint detection NN and utilizes a ResNet-18 encoder pretrained with ImageNet weights.

To optimize processing further, the original 4K image (3840 x 2160 pixels) is downsampled to 480 x 270 pixels, preserving the aspect ratio. However, to meet the stride requirement of 16 for ResNet-18, the image is cropped to 480 x 256 pixels. This ensures proper alignment while minimizing information loss. After detection, the center points are reverse-scaled and uncompressed to match the original 4K image workspace, maintaining consistency across resolutions.

4.3.2 Integration of Shaft Keypoint Detection

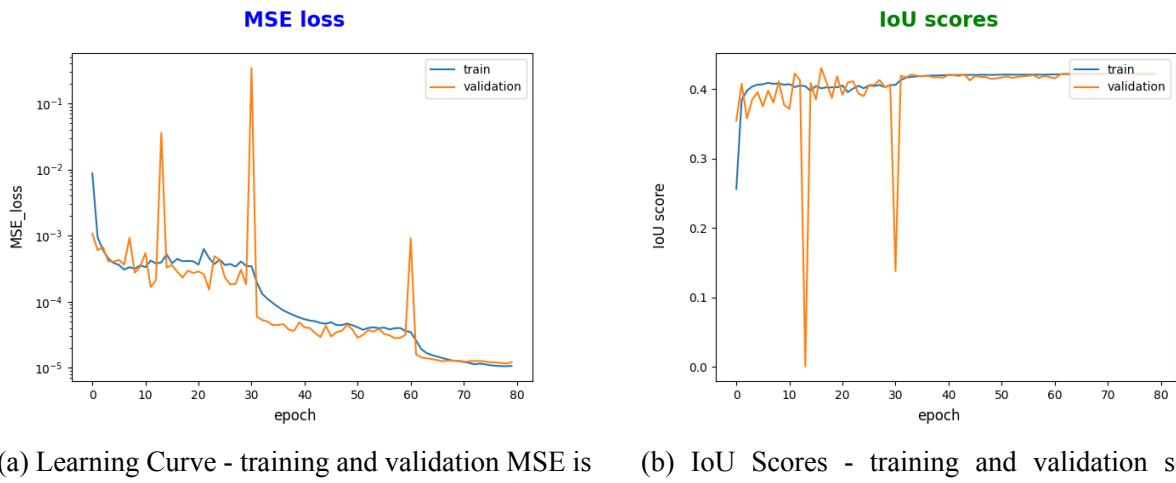
Initially, the surgical needle keypoint detection NN was also tasked with detecting the instrument shaft keypoint for the forceps. However, this approach encountered significant limitations. The hole created by the forceps arms and limited ROI size, made it inherently difficult to clearly distinguish the shaft keypoint.

Additionally, when the tip and shaft keypoints are close together, the arc angle between them becomes highly sensitive to small deviations in the shaft's detected position. Even minor errors in detecting the shaft's location can lead to significant inaccuracies in angle calculations. This issue can be mitigated by detecting the shaft keypoint farther along the instrument shaft, as this provides more stable angle measurements. This approach is better suited to a larger detection area, which the ROI NN provides.

As a result, the shaft keypoint detection was moved to the ROI NN. By leveraging the larger detection area of the ROI NN, the system achieves more consistent and accurate outputs for the shaft keypoint, particularly in complex surgical scenarios.

4.3.3 ROI NN Training Results

The resulting ROI NN is a robust system capable of accurately determining the ROI center point and size within the image workspace. The same training and validation is used as the main keypoint detection NN as described in Section 4.2.5. On the test set, the ROI model also achieved an MSE loss of 1×10^{-5} and an IoU score of around 0.4 as illustrated in Figure 4.5.



(a) Learning Curve - training and validation MSE is generalized well dropping to around 1×10^{-5} .

(b) IoU Scores - training and validation scores plateau around 0.4 with some fluctuations.

Fig. 4.5: ROI NN Model Training and Validation Results - the model converged successfully and minimally overfits.

4.3.4 Enhancements for Stability and Robustness

Smooth and stable ROI operation is crucial for accurate detection. To this end, the ROI NN incorporates simple moving average (SMA) and outlier prevention techniques. These methods stabilize the bounding box, reduce jittering, and ensure smoother tracking of instru-

ments, enhancing the user experience. The NN also limits ROI movement between frames, mitigating the effects of occasional misdetections. If an outlier is detected, the system retains the previous center point, preventing abrupt shifts or cascading errors.

In more challenging scenarios, such as when the forceps or shadow is near the camera's edge or hardware issues like camera freezing occur, the ROI may be misplaced. To address this, a user-configurable parameter allows the ROI to bypass outlier prevention after a set number of frames, enabling it to return to the correct position (see Figure 4.6). Without this function, the ROI could be inadvertently filtered out during its attempt to recover its position.

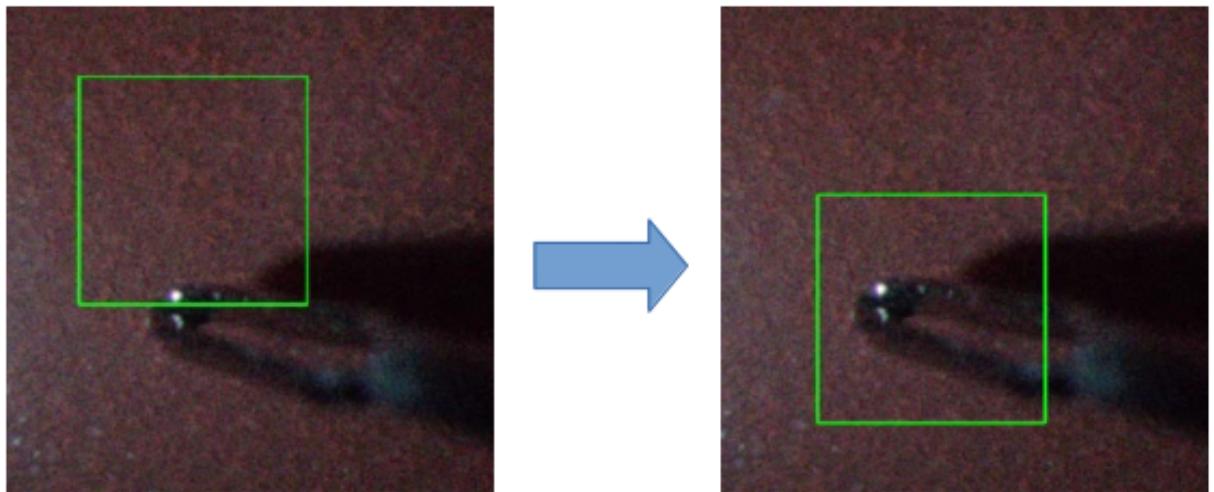


Fig. 4.6: The ROI system is capable of self-correction, even when the forceps tip and its shadow are positioned near or slightly outside the bounding box, by employing a recovery function after detecting frozen frames.

4.4 Advancements to the Autonomous Positioning Program

4.4.1 Autonomous Positioning Process Improvements

The autonomous positioning process underwent significant updates, integrating past versions with improvements to accommodate the current framework and kinematics. These updates not only enhance the system's functionality but also provide a robust foundation for future research, particularly for newer students or researchers entering the field.

The following improvements were implemented:

1. **Framework Integration:** The program's syntax and functions were updated to align with the SmartArmStack framework, as discussed in Section 3.3. This integration improves maintainability, streamlines development, and ensures compatibility with other programs and robotic systems within the lab.
2. **Optional Orbital Manipulation:** While orbital manipulation is unnecessary for the ILM peeling task, the updated program includes the kinematic functionality for future research purposes. A configurable setting allows orbital manipulation to be disabled while retaining its underlying kinematic calculations, enabling the program to support tasks that require orbital manipulation in conjunction with autonomous positioning.
3. **Simulation and Real Robot Compatibility:** The automation program manages the determination and calculation of target points, keypoints, and desired positions based on the operating environment. The program was advanced to operate seamlessly with both the simulation environment and real robots. This dual compatibility allows for efficient error testing in simulation before transitioning to physical robots, reducing potential risks.
4. **Code Refactoring for Maintainability:** Significant refactoring was performed to improve the program's readability and maintainability. A positioning helper function

class, as detailed in Section 3.3.2, consolidates repetitive calculations used across the six primary positioning steps. Additionally, a controller class was introduced to manage the sequence of the seven main steps for each target point. These changes reduce the program’s complexity, transforming it from nearly a thousand lines of code into a more modular and understandable structure.

These improvements ensure that the autonomous positioning program is not only more efficient and robust but also easier to adapt and expand for future applications.

4.4.2 Detecting the Target Points First

Detecting target points is a critical first step in the autonomous positioning workflow, and this process was restructured into a dedicated function. Once the target points are identified and confirmed, the surgical instruments are translated to their respective starting positions. Subsequently, keypoint detection and ROI initialization are performed. A picture of the target point detection before instrument translation is illustrated in Figure 4.7.

This separation of target point detection into an independent process optimizes resource usage by ensuring that this step is only executed once at the beginning of the program. It also streamlines the workflow, clearly delineating the stages where each detection process is required.

4.5 Integration of the Image Processing Node and Autonomous Positioning

The integration of the keypoint detection and ROI NNs into the robotic system required improvements to the architecture and workflow to support efficient and responsive operation. The image processing node plays a central role, managing the main capturing program, com-

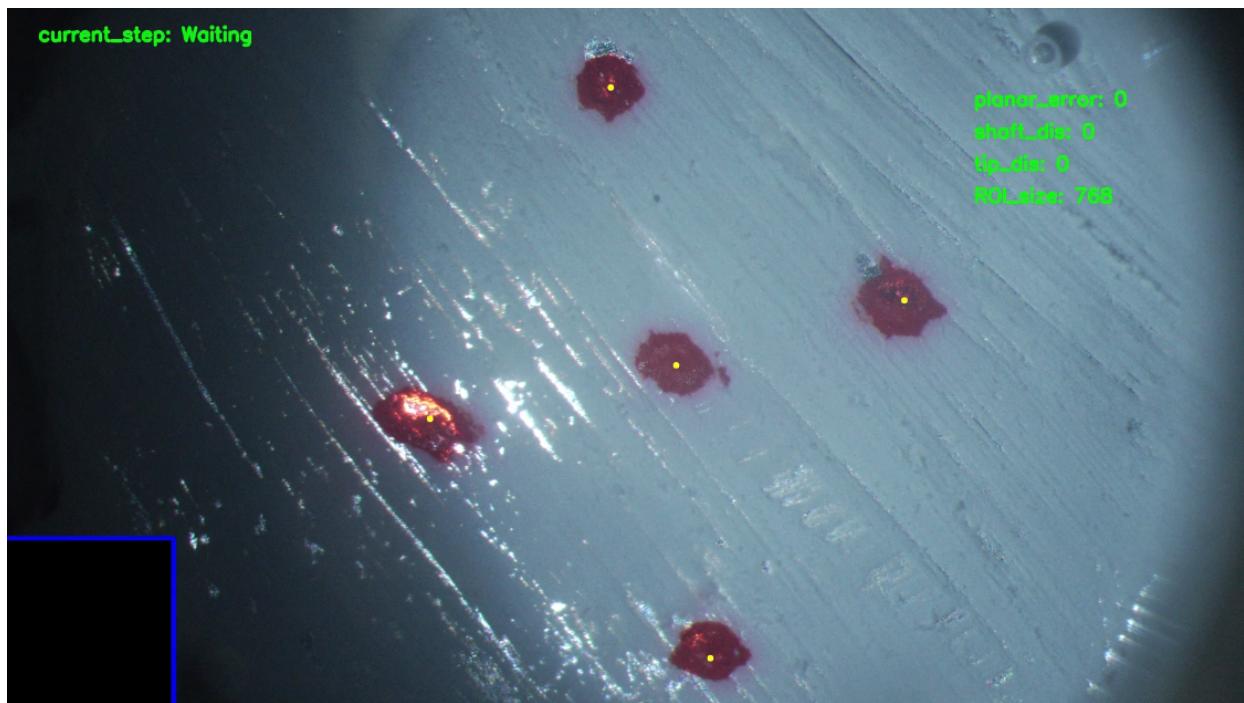


Fig. 4.7: Target point detection before instruments move to the starting position points.

municating with both NNs, and displaying marked video frames to the user. Meanwhile, the autonomous positioning program processes the detected keypoints and calculates the kinematics required to position the surgical instruments at the desired target points. Enhancements were introduced to streamline this workflow and optimize system resources.

4.5.1 Image Capturing Node Enhancements

The general process flow between the original surgical needle and the updated forceps image capturing node remains consistent; however, several changes and enhancements were implemented to accommodate the forceps and improve overall performance. The main capture program, written in C++, handles critical tasks such as processing video frames, detecting target points, managing publisher-subscriber communication between NNs, and marking up video frames for user observation.

The key changes and enhancements are as follows:

1. **Integration of the ROI Neural Network:** A new publisher and subscriber were introduced to facilitate communication with the ROI detection NN. Before sending images to the ROI NN, the main capture program performs workspace image compression and cropping. The program was also adapted to manage data exchange between the main keypoint detection NN and the ROI NN, ensuring seamless operation.
2. **Parallel Processing for Increased Speed:** Minor parallel processing was implemented to optimize runtime performance. Resizing images is computationally expensive due to pixel interpolation, memory access requirements, and the high resolution of the workspace images. By dividing these tasks among multiple processors, the workload is executed concurrently, maximizing the computational power of the system and reducing latency.
3. **Code Refactoring for Maintainability and Efficiency:** The original capture node suffered from outdated coding practices, leading to inefficiencies, potential bugs, and poor maintainability. Extensive refactoring addressed these issues:
 - ****Narrowing Conversions Prevention**:** Curly bracket '{ }' initialization is now used to prevent narrowing conversions, ensuring safer type assignments.
 - ****Memory Optimization**:** Pre-allocated memory addresses for frequently used objects and variables were introduced, reducing runtime overhead caused by repeated memory allocation.
 - ****Utilization of OpenCV Features**:** OpenCV's point data type was fully utilized to simplify and optimize image processing tasks.
 - ****Reorganization and Modularity**:** Repetitive code was replaced with loops and smaller, manageable functions, improving readability and modularity.
 - ****Static Constants and Templates**:** Constant image parameters were set as 'static constexpr' variables for compile-time evaluation and immutability. Templates were introduced to enhance code reusability, type safety, and flexibility.
 - ****Comprehensive Comments**:** Detailed comments were added throughout the codebase to improve maintainability and support for future developers.

These changes significantly improved runtime performance, reduced the potential for bugs, and enhanced the readability and scalability of the capture node.

4.5.2 Asynchronous Processing

Asynchronous processing was integrated into the system to minimize lag and ensure that keypoint detection operates on the most recent video frame. This improvement is especially important for maintaining accurate and responsive performance in dynamic surgical environments. Both the main keypoint detection NN and the ROI NN operate asynchronously, allowing them to work independently while remaining synchronized with the rest of the system.

When a neural network program receives an incoming message containing a compressed image, the frame is appended to a queue with a size of 1, ensuring that only the latest frame is retained. The program then processes the frame in the queue immediately, eliminating delays caused by outdated frames and maintaining real-time operation.

This architecture improves responsiveness and optimizes system resources by preventing the accumulation of unprocessed frames. Decoupling the two NNs ensures they operate in parallel, enabling each network to focus on its specific task—detecting keypoints or refining the ROI—without creating bottlenecks. The result is a more efficient and scalable system, capable of handling additional functionality in the future.

4.5.3 Architecture for Shadow-Based Autonomous Positioning

The architecture for shadow-based autonomous positioning integrates the image processing node and autonomous positioning program, forming the backbone of the system. These components work in tandem to facilitate real-time detection and precise instrument positioning.

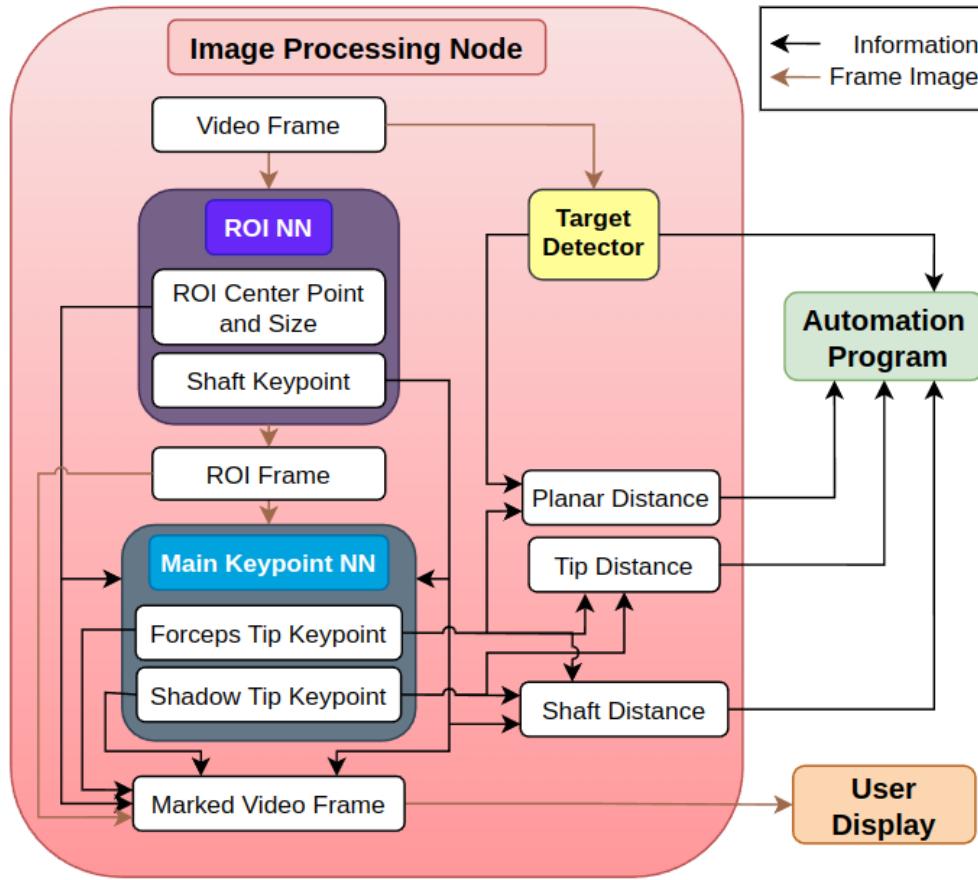


Fig. 4.8: Workflow of the Image Processing Node and Autonomous Positioning Program.

The image processing node captures video frames from the surgical workspace, applies preprocessing techniques, and interacts with the keypoint detection and ROI NNs. Leveraging asynchronous communication, the node ensures seamless integration between detection tasks and the robotic system. Detected keypoints and target points are processed to calculate positioning error, tip distance, and shaft distance. These values are then relayed to the autonomous positioning program, which computes kinematic solutions required to move the robotic instruments to their designated target positions. The overall workflow of the image processing node and autonomous positioning program is illustrated in Figure 4.8.

The system's modular design separates the detection and positioning processes, allowing continuous video frame processing while maintaining accurate and responsive robotic con-

trol. This decoupled architecture enhances system reliability and facilitates future updates or expansions, making it adaptable for increasingly complex image processing and surgical tasks.

4.6 Shadow-Based Autonomous Positioning with Forceps Experimental Setup

This experiment builds on the Autonomous Positioning Experiment with a surgical needle described in Section 2.2.5. While the previous experiment validated the capability of the Vitreoretinal SmartArm system for autonomous positioning with a simpler instrument, this study evaluates its feasibility with the more complex geometry of forceps.

4.6.1 Objectives

The primary objective is to demonstrate the system's capability to autonomously position the forceps tip at predefined target points on the fundus. This experiment will evaluate whether the system can effectively adapt to the forceps' more intricate geometry and dynamic orientation. The results will provide insights for further system improvements.

4.6.2 Methodology

The experiment involves autonomously positioning the forceps tip at five predefined target points on the fundus, sequentially ordered as p_1, p_2, p_3, p_4, p_5 . The four-step positioning strategy as described in Section 2.2.2 is used in conjunction with the forceps NN keypoint detection models. The experimental setup includes the following:

- **Eye Model and Fundus:**

- The eye model consists of an artificial fundus made from an aluminum sheet painted white to allow for electrical contact detection with little reflection.
- After two successful trials or six failed attempts, the fundus will be replaced to maintain consistent experimental conditions.

- **Forceps Positioning:**

- The robotic system is programmed to autonomously navigate to each target point using the shadow-based autonomous positioning algorithm.

- **Contact Detection:**

- An Arduino-based system is employed to detect contact between the forceps and the fundus.
 - Contact is determined through electrical continuity, ensuring precise confirmation of positioning. Force measurement is excluded in this experiment.

- **Calibration:**

- The robotic system is calibrated through an industry standard pivoting process [36].
 - The robots were calibrated for each respective fundus to ensure alignment between the robot's coordinate system and the eye model's geometry.

- **Trial Protocol:**

- Each trial involves the system attempting to position the forceps tip at all five target points.
 - A trial is considered successful if the system makes or attempts to make contact with all five target points, and no operator intervention is required to abort due to system failure.
 - A trial is aborted by the operator if any of the following conditions occur:
 - * The forceps continues translating into the fundus for more than two seconds.
 - * The forceps translates to a point outside the workspace instead of the target point.
 - * Image processing fails, resulting in unsafe system behavior.
 - * Other safety concerns, which will be mentioned in the results and discussion.
 - To ensure consistent wear-and-tear conditions and account for environmental

variations, the fundus is replaced after two successful trials or six failed trials, whichever occurs first.

- **Data Collection:**

- The system logs all of the configuration settings, robot kinematics, eye distance geometries, positioning information, and other information related to the autonomous positioning program such as distance errors and task duration.

4.6.3 Metrics for Evaluation

The metrics monitored in this experiment are: additional distance d_{add} (μm), positioning error E_p (μm), and task duration (s). Additional information about the significance and calculation of d_{add} and E_p is discussed in Section 2.2.2.

Additional Positioning

The final step in the automated positioning program, additional positioning, first calculates and then translates d_{add} the robot controller needs to translate down to the retinal surface. This experiment uses a threshold for the completion of vertical positioning of 15 pixels as compared to the 50 pixels:

$$d_{rest} [\mu\text{m}] = 15 [\text{px}] \times \frac{dz [\mu\text{m}]}{d_{xy} [\mu\text{m}]} \times \frac{1000}{417 [\text{px/mm}]} = 36 [\mu\text{m}] \times \frac{dz [\mu\text{m}]}{d_{xy} [\mu\text{m}]} \quad (4.6.1)$$

As a result, with the same positioning margin of $130 \mu\text{m}$, and d_{rest} of approximately $36 \mu\text{m}$, d_{add} should result in around $166 \mu\text{m}$.

Positioning Accuracy

The accuracy of positioning is evaluated using E_p , calculated from the recorded frame data. E_p is determined by measuring the distance between the forceps tip and the center of the red target point on the fundus (represented as a yellow dot in the UI). The recorded E_p value corresponds to the final iteration of the additional positioning step.

Task Duration

To evaluate efficiency, task duration is measured. The task duration can be used to compare between runs and the original experiment with the surgical needle.

4.6.4 Control and Comparison

The control for this experiment is the original autonomous positioning experiment conducted with a surgical needle, as described in Section 2.2.5. Both experiments utilize the Vitreoretinal SmartArm system and the shadow-based positioning methodology, ensuring a consistent framework for comparison. Key differences and control measures are outlined below:

- **Control Parameters:**

- The same surgical setup, besides the accommodation of the forceps, remains consistent between the two experiments.
- The four-step positioning strategy workflow is also consistent across both experiments.
- Metrics for evaluation, including E_p , d_{add} , and task duration, are identical to allow direct performance comparisons.
- The velocities and thresholds remain the same, unless noted below.

- **Key Experimental Differences:**

- **Instrument Geometry:** The surgical needle, a slender and symmetric instrument, is replaced with forceps, a more complex instrument with variable geometry and dynamic orientation.
- **Image Processing and Automation Architecture:** The forceps system utilizes the enhanced image processing node with the keypoint detection and ROI NNs. The image processing node communicates with the advanced autonomous positioning program.
- **Positioning Thresholds:**

- * In this experiment, the distance between the forceps tip and its shadow (d_{tip}) must reach or fall below 15 pixels (approximately $36 \mu\text{m}$) for at least 40 frames within a sliding window of 45 frames.
- * The sliding window function helps filter out false positives caused by temporary detection inaccuracies or noise.
- **Impact on Comparison:**
 - The forceps' increased complexity is expected to introduce additional challenges in maintaining positioning accuracy and minimizing task duration.
 - By comparing results between the surgical needle and forceps experiments, the study aims to quantify the system's adaptability and identify areas for improvement.

Chapter 5 Experimental Results and Discussion

This chapter presents the results of the shadow-based autonomous positioning with Forceps Experiment, including both quantitative and qualitative findings. The analysis focuses on the system's performance, drawing comparisons to the original experiment with the surgical needle. A detailed discussion follows, addressing key observations, challenges encountered, and the broader implications of the results.

5.1 Results

5.1.1 Quantitative Results

The experiment required twelve trials to meet the defined methodological goals. The first and third fundus each required three trials to achieve two successful runs, while the second fundus presented significant challenges, failing to complete a single run in six attempts. Overall, four trials were successfully completed, yielding a success rate of approximately 33% (4 out of 12). The durations for the successful trials were 410 s, 464 s, 409 s, and 267 s, with an average of 385 s and a standard deviation of 85 s. The quantitative results are summarized in Table 5.1. Box and whisker plots of additional positioning distance (d_{add}) and positioning

error (E_p) by target point is shown in Figure 5.1.

Trial #	fun	Additional distance d_{add} (μm),				Positioning Error E_p (μm)					
		p_1		p_2		p_3		p_4		p_5	
		d_{add}	E_p	d_{add}	E_p	d_{add}	E_p	d_{add}	E_p	d_{add}	E_p
Trial 1	1	187*	1167	197*	1027	—	—	—	—	—	—
Trial 2	1	181†	1162	191*	1059	184*	472	175*	809	185*	1038
Trial 3	1	183†	1120	194*	1042	181*	753	168†	589	177*	793
Trial 4	2	180†	1097	—	—	—	—	—	—	—	—
Trial 5	2	182†	1169	191*	968	—	—	—	—	—	—
Trial 6	2	180*	1107	196*	810	196*	497	181†	638	—	—
Trial 7	2	180*	1036	—	—	—	—	—	—	—	—
Trial 8	2	—	—	—	—	—	—	—	—	—	—
Trial 9	2	183*	1108	190*	807	185†	256	—	—	—	—
Trial 10	3	179*	1033	—	—	—	—	—	—	—	—
Trial 11	3	179*	1047	195*	964	184*	626	178†	722	198†	1024
Trial 12	3	180*	1001	184*	906	179*	762	172†	769	189*	950

* not positioned, † positioned, — force-quit

Table 5.1: Results of shadow-based autonomous positioning with forceps.

The system's ability to position the instrument was evaluated across 34 attempted target points, of which 10 were successfully positioned (†), resulting in a success rate of 29.4%. Success rates varied across target points: p_4 showed the highest rate at 80% with an average d_{add} of 175 μm , while p_2 had no successful attempts with an average d_{add} of 192 μm . The mid-range point, p_3 , demonstrated better accuracy, with an average E_p of $626 \pm 153 \mu\text{m}$, compared to peripheral points like p_1 with E_p of $1107 \pm 133 \mu\text{m}$. Figure 5.1b illustrates this observation with the mid-range point being the most accurate, followed by the left and right points, and then the bottom and top points. The E_p increased in inaccuracy the closer the target point was to the border of the microscopic workspace.

The system performed consistently when applying d_{add} , achieving a mean value of 183.2 μm with a standard deviation of 5.1 μm . However, the higher variability in E_p , with a mean of 904 μm and standard deviation of 291 μm , reflects challenges in lateral alignment, particularly at peripheral points such as p_5 .

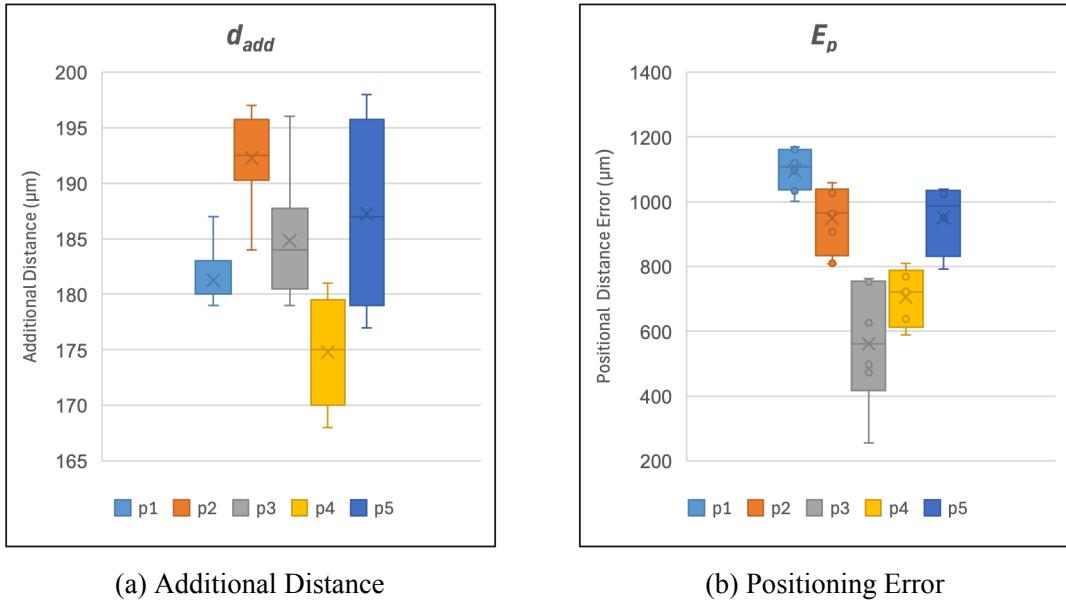


Fig. 5.1: Box and whisker plots of the five target points in the shadow-based autonomous positioning with forceps experiment.

5.1.2 Qualitative Observations

Several observations were made during the trials, highlighting both system performance and experimental challenges:

- **Keypoint Detection Challenges:**

- The keypoint detection occasionally struggled to locate the true position of the forceps tip and its shadow, particularly in specific orientations. Notable issues were observed during/after the overlap prevention step, vertical positioning, and at target point p_5 , located at the lower portion of the workspace (Figure 5.2).
- The top and bottom portions of the workspace occasionally had issues detecting the keypoints due to the ROI reaching the boundary. The instrument and shadow were not centered or sometimes not within the ROI anymore in these cases (Figure 5.3).
- As the forceps tip and shadow became closer, the keypoint detection of the shadow

tip became inaccurate when occluded by the forceps (Figure 5.4a).

- Vertical positioning sometimes ended early due to the shadow or tip being detected at the wrong pixel (Figure 5.4b).

- **Unsuccessful Trials:**

- Many unsuccessful trials were aborted due to excessive force applied by the forceps on the fundus. In some instances, the keypoint detection program failed to register the forceps tip and its shadow as being within the defined distance threshold, resulting in the manipulator continuing to translate into the fundus (Figure 5.5).

- **Sudden Instrument Movements:**

- The forceps exhibited sudden movements during some operations, which caused slight shifts in the entire workspace (Figure 5.6).

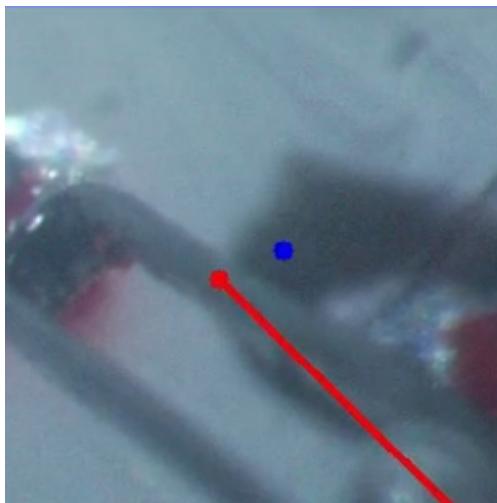
- **Video Frame Lag:**

- A delay of approximately two seconds was observed between robot movements and the corresponding marked video frames.

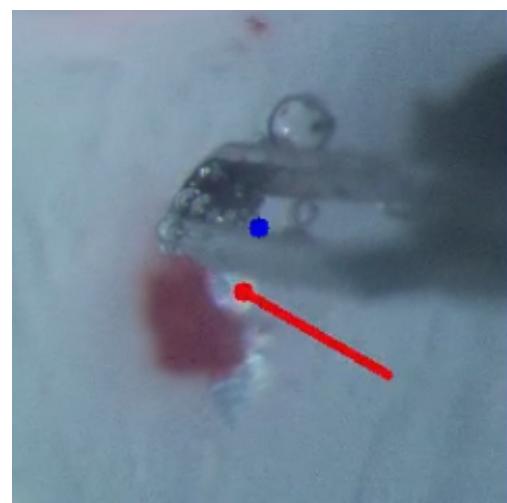
- **Fundus Wear and Environmental Factors:**

- As trials progressed, the fundus became increasingly scratched, with surface conditions varying across trials. Additionally, bubbles were observed to accumulate on the forceps and fundus surface (Figure 5.7).
- Occasionally, a bubble would float to the cornea, obscuring the image (Figure 5.8).
- The solution inside the eyeball was purified water, ruling out saline solution as a cause of increased conductivity. The larger surface area of the forceps may have contributed to higher conductivity and current within the setup.

Figures Related to Observations



(a) Forceps tip detection inaccuracy during overlap prevention step of p_3 .

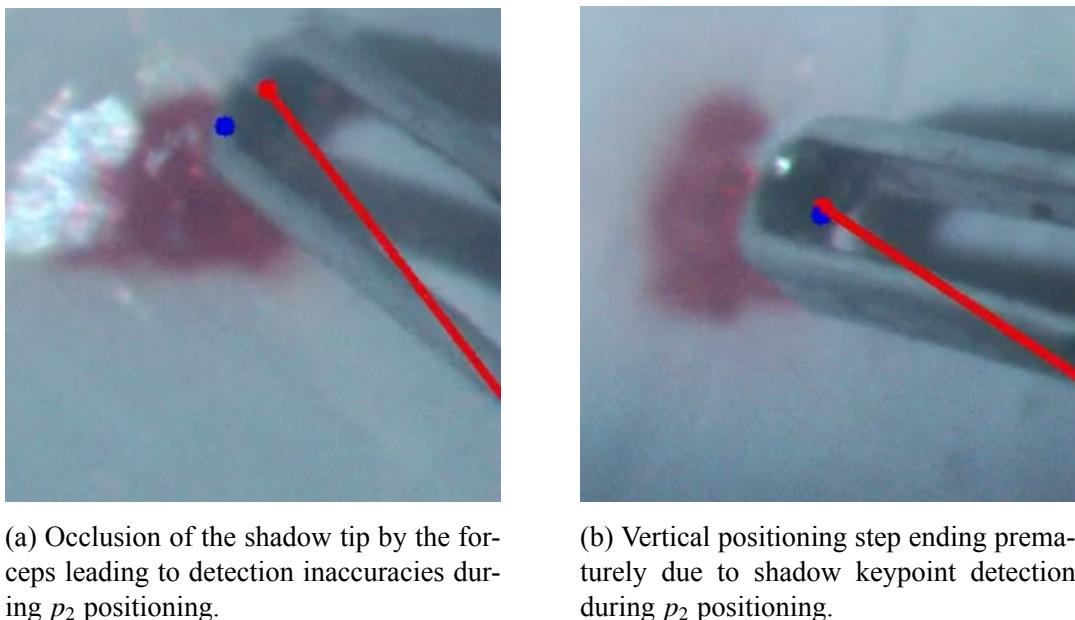


(b) Inaccuracies for forceps tip and shadow during the positioning of p_5 .

Fig. 5.2: Notable keypoint detection issues.



Fig. 5.3: ROI boundary issues affecting keypoint detection during p_1 positioning. The shadow is not within the ROI bounding box.



(a) Occlusion of the shadow tip by the forceps leading to detection inaccuracies during p_2 positioning.

(b) Vertical positioning step ending prematurely due to shadow keypoint detection during p_2 positioning.

Fig. 5.4: Minor inaccuracies for keypoint detection.

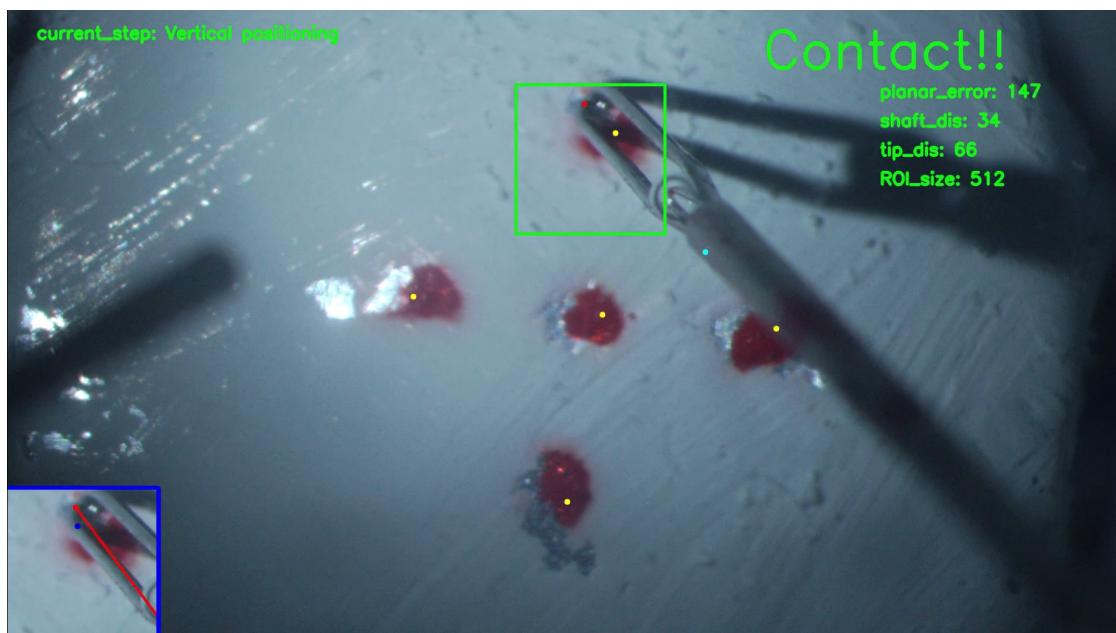


Fig. 5.5: Excessive force exerted on the fundus leading to aborting the program during vertical positioning of p_1 .

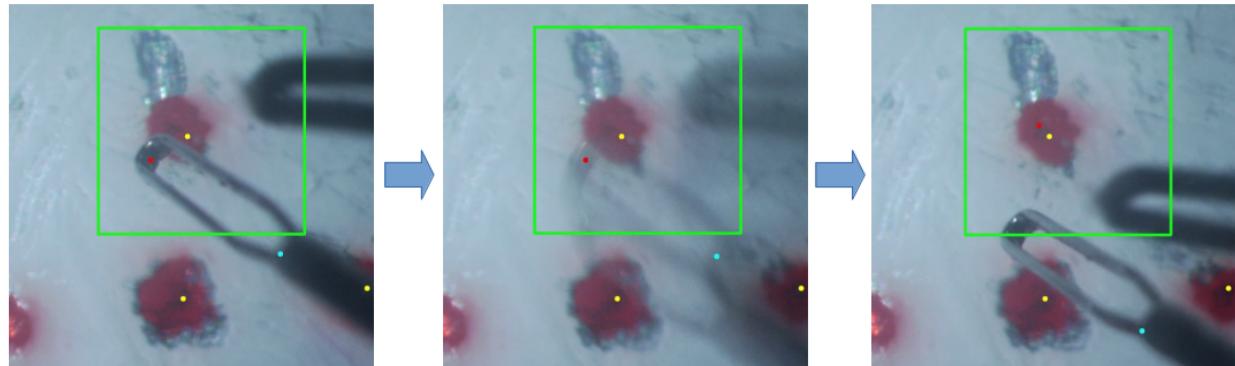
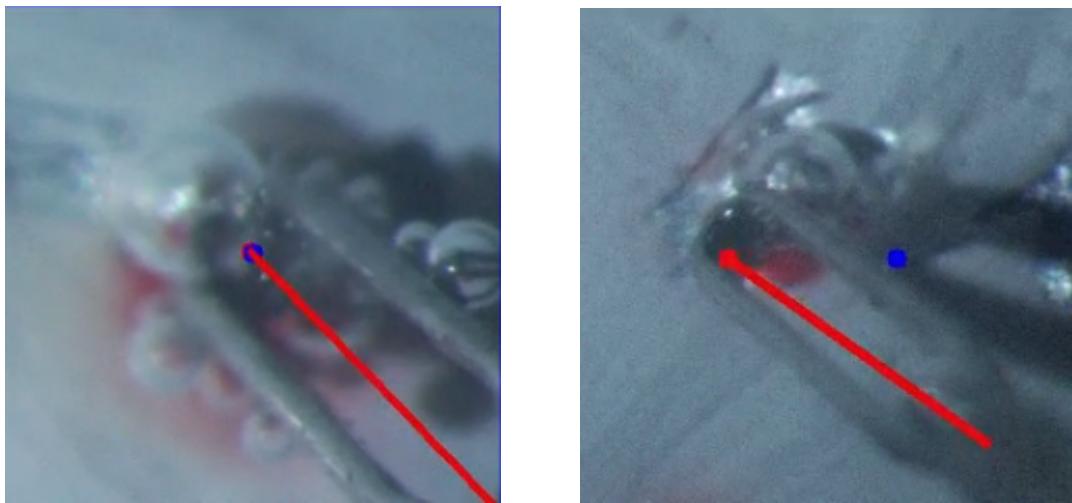


Fig. 5.6: Sudden instrument movements causing workspace shifts.



(a) Bubbles on the forceps and fundus during contact of p_4 .

(b) Excessive fundus wear and a bubble surrounding p_3 .

Fig. 5.7: Different environmental factors affecting image processing.

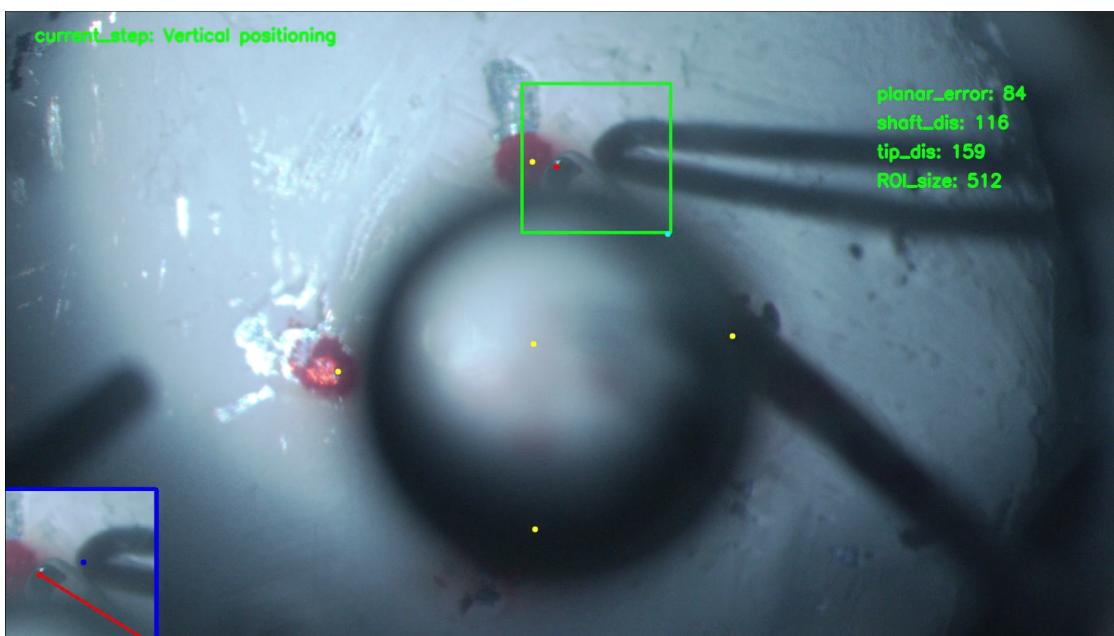


Fig. 5.8: Bubble on the cornea obscuring the image.

5.2 Discussion

The experiment successfully demonstrated the capability of shadow-based automated positioning with the updated forceps image processing and kinematic framework. However, with only four out of twelve trials considered successful, and ten out of the thirty-four targets correctly positioned, the results highlight significant areas for improvement in both system implementation and methodology.

5.2.1 Observations Analysis

Trial by Trial

Out of the twelve trials, only four were deemed successful, though these results can be considered borderline due to inconsistencies in keypoint detection. Many vertical positioning steps terminated prematurely, reflecting the need for more robust detection algorithms. Among the successful trials, only a few cases involved minimal force during the additional positioning step, indicating partial success in maintaining delicate interactions with the fundus.

Quantitative Results

The quantitative results underscore both the strengths and limitations of the current system. The consistent d_{add} values highlight the consistency of the additional positioning step, which plays a crucial role in precise instrument alignment. However, the vertical positioning step has high variability with many runs ending early or requiring abort. Additionally, the higher variability in E_p , particularly at peripheral points like p_5 , reveals the challenges in achieving accurate lateral alignment.

The differences in success rates across target points further illuminate these challenges. Mid-range points, such as p_3 , demonstrated better alignment accuracy, while peripheral points like p_1 and p_5 introduced greater difficulties. This discrepancy is likely due to issues in keypoint detection, shadow occlusion, and the increased complexity of positioning at the

workspace edges. These findings highlight the need for improved detection algorithms and positioning strategies to enhance the system's overall performance.

Sudden Movements

Sudden movements observed during the workspace transitions could stem from multiple factors, including inaccuracies in setting the remote center of motion (RCM). The instrument insertion length may have deviated from the assumed 5 mm, introducing slight orbital misalignments. Additionally, the eye model's 3D-printed concave mount, with its rough resolution and distinct ridges, likely contributed to occasional sudden shifts in the workspace.

Keypoint Detection

The image processing node's inconsistencies could be attributed to limitations in the training dataset and keypoint detection method. Adding more labeled images, especially those representing scenarios after overlap prevention and at the lower workspace boundaries, could improve detection accuracy. Including diverse backgrounds, varying levels of exposure, and bubble interference in training data would enhance the robustness of the NN. Furthermore, other methods to detect the shadow should be considered as the NN is struggling to correctly pinpoint the correct position when occluded behind the forceps.

Video Delay

Lastly, the observed two-second video frame delay significantly impacts the feasibility of semi-automated processes. While slow set velocities in this experiment helped mitigate the lag's effects, addressing this delay will be critical for future real-time applications.

5.2.2 Comparison to the Original Experiment

The comparison between the needle and forceps experiments highlights key performance differences, as summarized in Table 5.2.

Experiment	# of Trials	d_{add} (μm)	E_p (μm)	Duration (s)
Needle	5	310	600	257
Forceps	4	183	880	385

Table 5.2: Comparison of the mean values of successful trials for the needle and forceps experiments.

The surgical needle achieved better results, with more robust keypoint detection due to its simple and symmetric geometry, making it ideal for shadow-based positioning. The needle experiment required only seven trials to complete five successful runs, whereas the forceps' more complex geometry posed challenges.

The d_{add} values for the forceps were significantly lower than those for the needle as the vertical positioning threshold was set at 15 pixels instead of 50 pixels. Considering the theoretical distance for the needle should be 250 μm with the experimental d_{add} average of 210 μm , there is approximately a 19.4% error. Comparatively, the forceps resulted in 166 μm with the experimental d_{add} average of 183 μm , there is approximately a 10.2% error. This shows that the calibration of the forceps was more accurate for the forceps compared to the needle as this is based on a geometric relationship and not by image processing.

The E_p results highlight the needle's superior accuracy in positioning, attributed to its straightforward geometry. Despite both experiments being conducted at the same velocities, the forceps trials took significantly longer to complete.

One possible reason for the increased duration is that the starting position of the forceps may have been higher, necessitating a longer vertical positioning step. Furthermore, the forceps frequently overshot the target during planar positioning due to inferior keypoint detection accuracy. These overshoots required additional corrective movements, further increasing the overall task duration.

5.2.3 Challenges and Limitations

Several challenges and limitations impacted the experiment's outcomes:

- **Rotational Kinematics Bug:** As described in Section 3.2.2, a bug in the rotational kinematics likely contributed to inaccuracies during positioning. This may have caused subtle shifts in the RCM, impacting the overall precision. The change in workspace view can be observed in Figure 5.9.
- **Forceps Configuration:** The forceps remained closed throughout the experiment, which is suitable for planar and vertical positioning steps, but insufficient for tasks requiring interaction with the fundus. Proper configuration with partially opened forceps will be necessary for future experiments.
- **Simplified Environment:** The experimental setup, while effective for testing shadow-based positioning, does not accurately simulate the conditions of the Bionic-Eye production ILM model or a real eyeball. Even though the NNs are trained and compatible with the ILM model performance may vary significantly.
- **Fundus and Forceps Variability:** Wear on the fundus and bubble accumulation increased with successive trials, introducing inconsistencies that may have affected the results. This is quite evident with the second fundus failing all of its trials. Furthermore, even after cleaning the forceps before each trial, the forceps would accumulate bubbles and white paint from the fundus, which may have also affected the results.

5.2.4 Implications and Future Direction

The findings from this experiment underscore the potential of shadow-based automated positioning with forceps while revealing key challenges that must be addressed to develop a safer and more robust system. These challenges, as well as future directions for improving the Vitreoretinal SmartArm system, are contextualized within insights gained from the original needle-based experiment discussion [12]. With full automation of the ILM peeling process

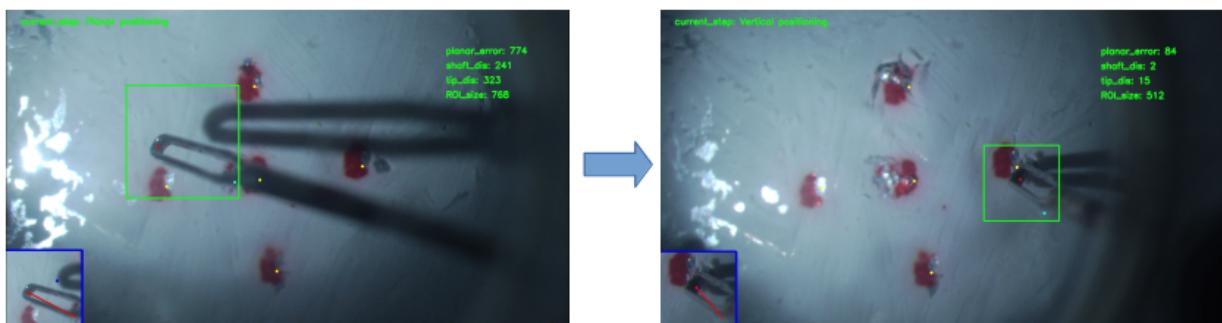


Fig. 5.9: The rotational kinematics bug can be seen on the right side of the workspace view. The eyeball slightly rotates without returning to center.

as the ultimate goal, achieving semi-automation serves as a crucial intermediate milestone in advancing robotic vitreoretinal surgery.

Future experiments should prioritize testing in more realistic environments, such as the Bionic-Eye production ILM model, to better align results with clinical applications. While this experiment was chosen for its similarity to the original needle positioning task, the limitations in its environment and forceps configuration must be addressed to advance the system toward clinical relevance. More specific future work is highlighted in the conclusion and future works chapter.

Chapter 6 Proposed Experiments and Methods

Internal limiting membrane (ILM) peeling is one of the most delicate procedures in vitreoretinal surgery, requiring precise manipulation of a thin retinal membrane only a few microns thick. The procedure poses significant challenges due to the confined workspace, minimal allowable forces, and the need for exceptional dexterity and precision. This chapter preliminarily explores a method to reduce excessive force applied on the fundus by aligning the forceps. Additionally, a feasibility study designed to evaluate the system's ILM peeling performance is proposed, offering a pathway for future advancements in robotic vitreoretinal surgery.

6.1 Preliminary Exploration for Forceps Alignment

6.1.1 Importance of the Alignment Process

Excessive force on the retina during ILM peeling is a critical concern, as it can lead to unintended damage and surgical complications. Consultations with expert surgeons from the University of Tokyo Hospital, Japan, revealed that nonoptimal rotational alignment of the forceps is a primary contributor to excessive force. Proper alignment ensures that both ends of the forceps tips make simultaneous contact with the fundus, distributing force evenly and minimizing the depth required for grasping.

As illustrated in Figure 6.1, nonoptimal alignment (Figure 6.1a) causes one arm of the forceps to contact the fundus before the other, increasing the risk of excessive force application. Optimal alignment (Figure 6.1b) allows both tips to touch the fundus simultaneously, reducing force concentration and improving surgical precision.

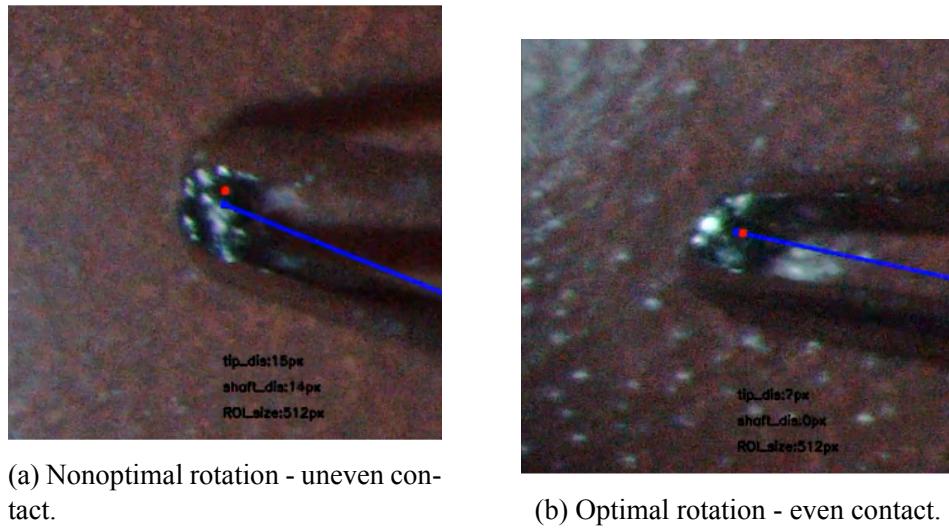


Fig. 6.1: Comparison of poor and optimal forceps alignment relative to the retina surface.

6.1.2 Future Development and Evaluation

The SmartArm system, with its advanced kinematics and control framework, is well-positioned to address forceps alignment challenges. Automated alignment methods leveraging precise rotational control can help achieve optimal forceps orientation before contact with the ILM, enhancing safety and surgical precision.

While the development and evaluation of a comprehensive alignment method were beyond the scope of this thesis, several aspects warrant further exploration:

- **Rotational Alignment Method:** Developing an algorithm to ensure accurate rotational positioning relative to the retina.

- **Rotation Calibration:** Investigating robust calibration methods for the forceps' rotational motor to ensure reliable alignment during operations.
- **Simulation and Validation:** Conducting experiments in both simulated and real surgical environments to assess alignment accuracy and its impact on force reduction.

This thesis provides a foundational framework for forceps alignment by integrating the necessary hardware and software systems. Future research should build on this work to refine and validate alignment methodologies, ensuring safer and more precise robotic-assisted ILM peeling.

6.2 Experimental Plan for ILM Peeling

Due to time constraints, the ILM peeling experiment will be conducted after the submission of this thesis. This experiment is designed to evaluate the feasibility of performing the ILM peeling task using the teleoperative Vitreoretinal SmartArm system integrated with the forceps device.

6.2.1 Experiment Overview

The experiment will employ three Bionic-Eye production ILM models, which limits the number of trials that can be conducted. Given the high value and limited availability of these models, only experienced surgeons will perform the task. The system's autonomous light guide feature will be activated to enable single-handed operation of the forceps during the procedure.

The type of the primary forceps instrument is usually selected based on the surgeon's preference. To ensure robustness and precision, a surgeon from the University of Tokyo Hospital, Japan, recommended exploring the use of the Alcon® Grieshaber® Maxgrip Forceps 25

Ga+^{*1}, a forceps instrument with a pointed tip and no central gap as shown in Figure 6.2. This instrument, offers improved keypoint detection potential due to its distinctive geometry. Future studies could benefit from incorporating this tool into the system, as it may simplify NN implementation for forceps tip and shadow detection. Furthermore, the recordings from this experiment will serve as a valuable resource for creating a dataset to train future NNs utilizing this forceps type.



Fig. 6.2: The Alcon® Grieshaber® Maxgrip Forceps 25 Ga+, a pointed, center-less forceps device.

Surgeon Preparation

To ensure familiarity with the system and minimize variability, surgeons will undergo a structured preparation process:

- **Simulator Training:** Our lab has developed a VR simulator capable of performing ILM ablation as shown in 6.3. The surgeons will first practice the ILM peeling task using this simulator. This step allows them to become accustomed to the haptic devices and their response during teleoperation.
- **Practice with a Used ILM Model:** Following simulator training, surgeons will teleoperate the real robotic system to perform practice sessions on a previously used Bionic-Eye production ILM model. This phase helps the surgeons adapt to the robotic controls, the surgical environment, and the specific characteristics of the ILM model.
- **Final Procedure:** The surgeons will perform the ILM foveal-sparing procedure on the three available Bionic-Eye production ILM models.

^{*1}<https://www.alconnordicsurgicalproducts.com/node/776>

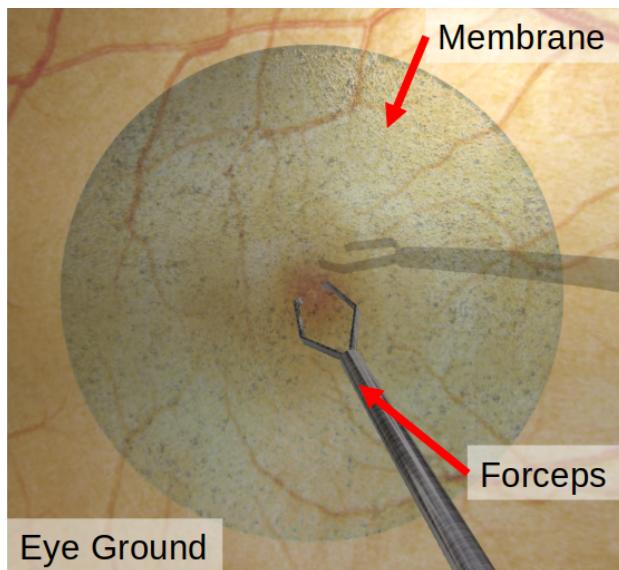


Fig. 6.3: ILM peeling task simulator developed by our group.

Evaluation Metrics

The results of the ILM peeling task will be assessed using the following methods:

- **Surface Deformation Analysis:** A stereo vision technique developed in the lab will be employed to measure observable surface deformation. This technique ensures an objective evaluation of the precision and effectiveness of the peeling task.
- **NASA Task Load Index (NASA-TLX):** Surgeons will complete the NASA Task Load Index after performing the task. This subjective tool evaluates the participant's mental workload (MWL) and is well-suited for assessing the cognitive and physical demands of surgical tasks.

6.2.2 Anticipated Challenges

Based on feedback from previous experiments with expert surgeons using the Vitreoretinal SmartArm system, as discussed in Section 2.2.5, several challenges have been identified that could affect the performance and usability of the setup during ILM peeling tasks.

One of the primary limitations is the absence of integrated haptic feedback, which makes it difficult for surgeons to sense the delicate interaction between the forceps and the ILM. Additionally, without tactile cues, maintaining awareness of the instrument's distance from the fundus becomes significantly more challenging. These limitations may lead to increased reliance on visual feedback, which can strain cognitive resources, particularly in a highly precise and constrained workspace.

Another anticipated challenge is the potential learning curve for operating the robotic system. Despite practice sessions, the time allotted for surgeons to acclimate to the system may be insufficient for them to become fully accustomed to its controls and nuances. This could affect their confidence and precision during the procedure.

Additional issues that may arise include:

- **Forceps Control:** The button mechanism for opening and closing the forceps on the haptic device may not feel intuitive compared to more natural controls, such as a pinching haptic device. This could lead to inefficiencies or errors during manipulation.
- **Visual Limitations:** The ILM may be difficult to distinguish from the background due to reflections, glare, or limited contrast, making it harder to visually confirm contact or peeling progress.
- **Stereo Vision Constraints:** The stereo vision system, as implemented through a microscope, may encounter difficulties with accurate depth perception in such a microscopic and highly precise environment.
- **Network Delay with Forceps:** The forceps device is controlled on a standalone computer, which is not ideal as there is a communication delay between the central computer and forceps computer.

6.2.3 Expected Outcomes

The ILM peeling task is expected to demonstrate the feasibility of using the Vitreoretinal SmartArm system with the forceps device in a teleoperative setting. The following outcomes are anticipated:

- **Successful ILM Peeling:** Surgeons will successfully complete the ILM foveal-sparing peeling task on the Bionic-Eye production ILM models, demonstrating the system's capability to perform delicate, precision-driven tasks.
- **Surgeon Performance Variability:** Variability in performance metrics (e.g., task duration, surface deformation) is expected due to the learning curve, limited practice time, and limited ILM models.
- **Workload Insights:** NASA-TLX scores will provide insights into the cognitive and physical demands of the task, highlighting areas where system usability can be improved.
- **Identified Challenges:** Challenges discussed in the anticipated challenges section, such as issues with forceps control, visual limitations, and stereo vision depth perception, may manifest, providing valuable feedback for future system refinements.
- **Quantifiable Metrics:** Surface deformation analysis and workload evaluations will establish a baseline for measuring the precision and usability of the system.

6.3 Discussion

The expected success of the ILM peeling task would validate the feasibility of using the Vitreoretinal SmartArm system with a forceps device for delicate surgical tasks. However, the observed outcomes will likely reveal limitations and provide critical insights for further development:

- **System Usability and Surgeon Adaptation:** Variability in surgeon performance may indicate the need for longer or more tailored training programs. Improving the intu-

itiveness of the forceps control mechanism, such as implementing a pinching haptic device, could enhance usability and reduce the learning curve.

- **Impact of Visual and Feedback Limitations:** Difficulties in distinguishing the ILM from the background or achieving accurate depth perception may emphasize the need for visual processing enhancements, such as preimage processing to improve contrast and reduce glare for the operator.
- **Stereo Vision and Depth Perception:** If stereo vision cannot accurately discern surface deformation, addressing microscope alignment and calibration issues will be crucial for achieving better accuracy.
- **Workload Insights:** High NASA-TLX scores in specific dimensions, such as mental or physical demand, may highlight system components that require optimization to reduce surgeon fatigue.
- **Future Improvements:** The anticipated challenges and their impact on performance metrics will guide the refinement of system features, such as haptic feedback integration, enhanced image processing, and more robust autonomous features.

These insights will not only improve the current system for the ILM peeling task but also contribute to others tasks and the broader development of autonomous robotic systems for precision surgical tasks.

Chapter 7 Conclusion and Future Work

This thesis focused on the integration of a forceps device into the Vitreoretinal SmartArm system, with the overarching goal of advancing automation in forceps positioning by modifying the existing framework and pipeline.

This chapter summarizes the key findings, highlights the primary contributions of this research, and outlines potential directions for future advancements in robotic vitreoretinal surgery.

7.1 Conclusions

This section provides conclusions for each main chapter of the thesis that contains contributions.

7.1.1 Integration of the Forceps

The Vitreoretinal SmartArm system has undergone significant enhancements to support the integration of a forceps device, enabling precise execution of the ILM peeling surgical task. Building upon proven hardware and software architectures established in previous research, the system has been adapted to effectively manage and operate the forceps device within the

constraints of vitreoretinal surgery.

The forceps device, including its robotic attachment, control box, and computer hardware, was meticulously designed to ensure accuracy and reliability in the highly restricted surgical workspace. Testing in a simulated environment provided a safe platform to validate the integration and operation of the real robots with the forceps, minimizing risks during deployment.

In addition to hardware advancements, substantial software improvements were made to enhance the system's functionality and maintainability. Key updates include the transition to the SmartArmStack ROS1 framework, the refactoring of hard-coded parameters to accommodate additional degrees of freedom, and the introduction of a helper function class to streamline robot positioning and optimization tasks. These refinements ensure the system is more robust, adaptable to future developments, and easier to maintain.

Together, these improvements position the Vitreoretinal SmartArm system as a reliable, state-of-the-art solution for performing ILM peeling, addressing both the precision required in surgery and the technical challenges of robotic integration.

7.1.2 Shadow-Based Autonomous Positioning with Forceps

The shadow-based autonomous positioning for use with forceps was implemented, building upon the existing framework originally designed for a surgical needle. The modifications to the image processing node were critical in addressing the unique challenges posed by the forceps' complex geometry and dynamic orientations. The updated image processing node incorporated enhancements tailored for forceps, including improvements to keypoint detection, the introduction of an additional NN to output an ideal location for both the ROI and shaft point, and refactored code. These updates set the foundation for the shadow-based autonomous positioning with forceps experiment and future work with orbital manipulation.

The shadow-based automation program was advanced for compatibility with the simula-

tion and real robots with increased maintainability and efficiency. The automation program seamlessly communicates with the image processing node, allowing the robot manipulators to orientate and translate to their desired positions.

7.1.3 Experimental Results and Discussion

Despite the advancements in forceps keypoint detection, the system's performance highlighted key limitations, such as inconsistencies in keypoint detection during certain orientations, challenges with vertical positioning, and increased task difficulty caused by environmental factors like bubble accumulation and occlusion of the shadow by the forceps.

Comparisons to the original needle-based experiment revealed that the forceps' geometry significantly impacts system accuracy, as evidenced by higher E_p values and lower success rates. This highlights the limitations in the existing shadow-based positioning framework. These differences highlight the adaptability required for handling more complex instruments compared to the simpler geometry of the surgical needle.

While the experiment successfully demonstrated the feasibility of adapting shadow-based positioning for forceps, it also underscored the need for further refinements. These advancements are discussed in the subsequent future works section.

7.1.4 Proposed Experiments and Methods

This chapter emphasized the critical importance of forceps alignment in achieving safe and effective ILM peeling. Although time constraints limited the full implementation of the alignment method, the preliminary exploration provided a strong foundation for future work. The method aims to optimize the rotational positioning of the forceps to ensure that both tips make simultaneous contact with the fundus. This alignment minimizes the depth required for grasping, reduces the risk of excessive force application, and leverages the system's advanced

kinematics for enhanced safety and precision.

Additionally, a structured experimental plan was proposed to evaluate the feasibility of the system for ILM peeling using Bionic-Eye models. The plan includes surgeon preparation through simulation training and practice on used ILM models, followed by evaluation on high-fidelity Bionic-Eye models. Quantitative metrics such as surface deformation analysis and subjective metrics like NASA-TLX scores were introduced to provide a comprehensive assessment of system performance. Challenges identified during the planning phase, such as visual limitations and the absence of haptic feedback, were acknowledged as areas for future improvement.

These contributions lay the groundwork for validating the system's capabilities in ILM peeling, offering insights into both the current limitations and the potential for future advancements in robotic-assisted vitreoretinal surgery.

7.2 Future Work

1. **Rotational Kinematics Bug:** Resolving this bug, detailed in Section 3.2.2, will help mitigate inaccuracies during positioning. It is also necessary to address before implementing orbital manipulation with the forceps device.
2. **Improving the Hardware setup:** The forceps computer should be integrated into the central computer for reduced latency between the operator and forceps device. The central computer requires a better CPU and other possible upgrades to accommodate more advanced NNs.
3. **Dataset Expansion:** Expanding the dataset with a broader range of images is critical to enhancing the NN's robustness. Priority should be given to incorporating images with diverse backgrounds, lighting conditions, instrument orientations, and edge-of-workspace scenarios. Emphasis on the Bionic-Eye ILM model will be essential for evaluating ILM peeling feasibility under realistic conditions.

4. **Optimizing the Image Detection Node:** The image processing workflow can be improved further to determine the two-second video delay. Investigating which pre-image processing techniques are the most effective can improve keypoint detection and processing load.
5. **Advanced Image Techniques:** There is human error when labeling images in this keypoint detection method as the desired points can be hard to discern. Transitioning from keypoint detection to image segmentation could improve accuracy for forceps detection. Amodal segmentation may also be useful for handling occluded shadows.
6. **Forceps Rotational Alignment Method:** A method to reduce the force exerted on the retina by optimally aligning both arms of the forceps is described in Section 6.1.
7. **ILM Peeling with Vitreoretinal SmartArm system:** The ILM peeling experiment is outlined with expected challenges and outcomes in Section 6.2. The feasibility of the forceps system will be demonstrated.
8. **Semi-Automated Processes:** Developing intermediate automation steps, such as automated vertical positioning, could enhance safety and reduce the cognitive load for surgeons. This is especially important for the initial grasp and tear of the ILM.

Chapter 8 Appendix

8.1 Dual quaternions and operators

Dual quaternions are a powerful mathematical tool for encoding spatial information of objects within a three-dimensional space. One of the primary advantages of dual quaternions is their efficient and compact representation. Unlike homogeneous transformation matrices [41], which also describe spatial transformations, dual quaternions require only eight parameters instead of the sixteen required by homogeneous matrices. This compactness not only simplifies calculations but also enhances computational efficiency. Additionally, dual quaternions possess several intriguing properties that make them particularly well-suited for robotic applications. These properties will be explored in greater detail in the following sections.

Dual quaternion algebra offers significant advantages over other representation methods. Unit dual quaternions, in particular, are free from representational singularities and are more compact and computationally efficient than homogeneous transformation matrices. Their robust algebraic properties enable systematic modeling of various robotic systems. Furthermore, dual quaternions allow for the straightforward representation of rigid motions, twists, wrenches, and a variety of geometric primitives, including Plücker lines and planes [42].

8.1.1 The dual number set, \mathbb{D}

Dual numbers were originally proposed by Clifford [43], which introduced the nilpotent dual unit with the following properties

$$\epsilon \neq 0, \epsilon^2 = 0. \quad (8.1.1)$$

Similar to complex number theory, the dual operator ϵ is used to break the problem into two components. The set of dual numbers is define as

$$\mathbb{D} \triangleq \{\underline{a} = a + \epsilon a' : a, a' \in \mathbb{R}, \epsilon \neq 0, \epsilon^2 = 0\}.$$

While dual number theory can be extended to various other concepts, this section specifically concentrates on their application in combination with quaternions to represent rigid motions.

To facilitate the use of dual numbers, it is useful to define operators that extract the primary and dual components of a dual number. We define an operator that extracts the dual component, which follows the dual unit, as

$$\mathcal{D}(\underline{a}) \triangleq a'$$

and the primary part, which does not follow the dual unit, as

$$\mathcal{P}(\underline{a}) \triangleq a.$$

Any sum or multiplication of dual numbers^{*1} has only to respect (8.1.1), and any remaining operations follow from common algebra. For instance, for $\underline{a}, \underline{b} \in \mathbb{D}$, we have

$$\underline{a} \pm \underline{b} = (a \pm b) + \varepsilon(a' \pm b'),$$

which is quite direct. The properties of the dual unit are visible in multiplication, in which we have

$$\begin{aligned}\underline{a}\underline{b} &= \underline{b}\underline{a} = ab + \varepsilon(ab' + a'b) + \varepsilon^2 a'b' \xrightarrow{0} \\ &= ab + \varepsilon(ab' + a'b).\end{aligned}$$

8.1.2 The quaternion set, \mathbb{H}

Quaternions were formulated by W. R. Hamilton [44]. They can be seen as an extension of imaginary numbers to four degrees-of-freedom with the imaginary units having the following properties:

$$\hat{i}\hat{j}\hat{k} = \hat{i}^2 = \hat{j}^2 = \hat{k}^2 = -1, \quad (8.1.2)$$

from which 1 is, as in \mathbb{R} , the identity

$$1^2 = 1. \quad (8.1.3)$$

^{*1}The division by dual units is out of the scope of this research.

Moreover, from following this property, we have

$$\begin{aligned}\hat{i}\hat{j} &= -\hat{j}\hat{i} = \hat{k}, \\ \hat{j}\hat{k} &= -\hat{k}\hat{j} = \hat{i}, \\ \hat{k}\hat{i} &= -\hat{i}\hat{k} = \hat{j}.\end{aligned}$$

The set of quaternions can be then formally defined as

$$\mathbb{H} \triangleq \left\{ h_1 + \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4 : h_1, h_2, h_3, h_4 \in \mathbb{R} \right\}.$$

Any operation with quaternions then needs only to respect (8.1.2) and (8.1.3), aside divisions by imaginary numbers which are out of the scope of this thesis. For $\mathbf{h}, \mathbf{g} \in \mathbb{H}$, we have

$$\mathbf{h} \pm \mathbf{g} = (h_1 \pm g_1) + \hat{i}(h_2 \pm g_2) + \hat{j}(h_3 \pm g_3) + \hat{k}(h_4 \pm g_4)$$

and

$$\begin{aligned}\mathbf{h}\mathbf{g} &= (h_1g_1 - h_2g_2 - h_3g_3 - h_4g_4) + \\ &\quad \hat{i}(h_1g_2 + h_2g_1 + h_3g_4 - h_4g_3) + \\ &\quad \hat{j}(h_1g_3 + h_2g_4 + h_3g_1 + h_4g_2) + \\ &\quad \hat{k}(h_1g_4 + h_2g_3 - h_3g_2 + h_4g_1).\end{aligned}$$

It is important to note that in general $\mathbf{h}\mathbf{g} \neq \mathbf{g}\mathbf{h}$.

Operators to retrieve the real and imaginary parts of a quaternion often are proven useful when describing other properties. To retrieve the real part of a quaternion we define the operator

$$\text{Re}(\mathbf{h}) \triangleq h_1,$$

and to retrieve the imaginary part we define the operator

$$\text{Im}(\mathbf{h}) \triangleq \hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4.$$

The quaternion conjugate can be defined as

$$\mathbf{h}^* \triangleq \text{Re}(\mathbf{h}) - \text{Im}(\mathbf{h}),$$

which is the quaternion in which the imaginary part has a flipped sign. The conjugate itself is convenient for defining the quaternion norm

$$\|\mathbf{h}\| \triangleq \sqrt{\mathbf{h}^* \mathbf{h}} = \sqrt{\mathbf{h} \mathbf{h}^*} = \sqrt{\sum_{n=1}^4 h_n^2} \in \mathbb{R}. \quad (8.1.4)$$

A combination of those properties gives us the quaternion inverse. For a non-zero quaternion, $\mathbf{h} \in \mathbb{H} - \{0\}$, its inverse is given by

$$\mathbf{h}^{-1} = \frac{\mathbf{h}^*}{\|\mathbf{h}\|^2}. \quad (8.1.5)$$

The proof follows from the quaternion norm definition (8.1.4). If \mathbf{h}^{-1} is the inverse of \mathbf{h} , then,

$$\mathbf{h} \mathbf{h}^{-1} = 1 = \mathbf{h}^{-1} \mathbf{h}.$$

From the first equality, we have

$$\begin{aligned} \mathbf{h} \mathbf{h}^{-1} &= 1 \\ \mathbf{h}^* \mathbf{h} \mathbf{h}^{-1} &= \mathbf{h}^* \\ \|\mathbf{h}\|^2 \mathbf{h}^{-1} &= \mathbf{h}^* \\ \mathbf{h}^{-1} &= \frac{\mathbf{h}^*}{\|\mathbf{h}\|^2}. \end{aligned}$$

The reasoning for the second equality is the same.

When it exists, the time-derivative of the squared norm of a time-varying quaternion $\mathbf{h}(t) \in \mathbb{H}_p$ is given by

$$\frac{d}{dt} (\|\mathbf{h}\|^2) = \dot{\mathbf{h}} \mathbf{h}^* + \mathbf{h} \dot{\mathbf{h}}^* = 2 \langle \dot{\mathbf{h}}, \mathbf{h} \rangle. \quad (8.1.6)$$

8.1.3 The pure quaternion set, \mathbb{H}_p , and vectors in \mathbb{R}^3

\mathbb{R}^3 can be defined in terms of the imaginary units as the pure quaternion set

$$\mathbb{H}_p \triangleq \{\mathbf{h} \in \mathbb{H} : \operatorname{Re}(\mathbf{h}) = 0\},$$

in which the subscript p stands for pure. This way, a quaternion $\mathbf{v} = x\hat{i} + y\hat{j} + z\hat{k} \in \mathbb{H}_p$ represents the vector $(x, y, z) \in \mathbb{R}^3$. Notice that for $\mathbf{u} \in \mathbb{H}_p$,

$$\begin{aligned}\mathbf{u}^* &= \operatorname{Re}(\mathbf{u}) - \operatorname{Im}(\mathbf{u}) \\ &= -\operatorname{Im}(\mathbf{u}) \\ &= -\mathbf{u}.\end{aligned}\tag{8.1.7}$$

For $\mathbf{u}, \mathbf{v} \in \mathbb{H}_p$, the inner product can be obtained as

$$\langle \mathbf{u}, \mathbf{v} \rangle \triangleq -\frac{\mathbf{u}\mathbf{v} + \mathbf{v}\mathbf{u}}{2} = \sum_{n=1}^4 (u_n v_n)^2 \in \mathbb{R}.\tag{8.1.8}$$

It is useful to notice the relation between the inner product (8.1.8) and the norm (8.1.4)

$$\langle \mathbf{u}, \mathbf{u} \rangle = \sum_{n=1}^4 u_n^2 = \|\mathbf{u}\|^2.\tag{8.1.9}$$

The cross product is defined as

$$\begin{aligned}\mathbf{u} \times \mathbf{v} &= -\mathbf{v} \times \mathbf{u} \triangleq \frac{\mathbf{u}\mathbf{v} - \mathbf{v}\mathbf{u}}{2} \\ &= \hat{i}(u_3 v_4 - u_4 v_3) + \hat{j}(u_4 v_2 - u_2 v_4) + \hat{k}(u_2 v_3 - u_3 v_2) \in \mathbb{H}_p.\end{aligned}\tag{8.1.10}$$

8.1.4 The unit quaternion set, \mathbb{S}^3 , and rotations

The set of quaternions with unit norm,

$$\mathbb{S}^3 \triangleq \{\mathbf{h} \in \mathbb{H} : \|\mathbf{h}\| = 1\},$$

represent the rotation,

$$\mathbf{r} = \cos(\phi/2) + \mathbf{v} \sin(\phi/2),\tag{8.1.11}$$

where $\phi \in \mathbb{R}$ is the clockwise rotation angle about the rotation axis $\mathbf{v} \in \mathbb{S}^3 \cap \mathbb{H}_p$. This means that any unit norm quaternion represents a rotation of an arbitrary angle about an arbitrary axis. The opposing rotation is obtained by flipping the sign of \mathbf{v} , which corresponds to

$$\mathbf{r}^* = \cos(\phi/2) - \mathbf{v} \sin(\phi/2),$$

which is the clockwise rotation of ϕ about the rotation axis $-\mathbf{v}$, which is the counter-clockwise ϕ about the rotation axis \mathbf{v} . Hence, it is only expected that

$$\mathbf{r}\mathbf{r}^* = \mathbf{r}^*\mathbf{r} = 1, \quad (8.1.12)$$

since a rotation followed by its exact opposite returns us to the same place. This also corroborates with the quaternion inverse (8.1.5). For $\mathbf{r} \in \mathbb{S}^3$

$$\mathbf{r}^{-1} = \frac{\mathbf{r}^*}{\|\mathbf{r}\|^2} = \mathbf{r}^*.$$

It is important to note that both \mathbf{r} and $-\mathbf{r}$ represent the same rotation, a phenomenon known as the quaternion double cover problem. This issue can be addressed by adjusting the error, as discuss in detail in Section 8.3, in our controller.

A rotation such as (8.1.11) is always relative to a reference frame, \mathcal{F}_a , moving the object either to a new reference frame or back to the frame itself. The rotation of a frame \mathcal{F}_a with respect to itself is always

$$\mathbf{r}_a^a = 1. \quad (8.1.13)$$

Any further rotation can be obtained by right-side multiplication of a rotation quaternion. To rotate from \mathcal{F}_a to \mathcal{F}_b , in the point of view of \mathcal{F}_a

$$\mathbf{r}_a^a \mathbf{r}_b^a = \mathbf{r}_b^a,$$

in which the superscript is the reference frame, and the subscript is the resulting frame. Using this notation, a relative rotation always requires a pairing between the subscript of the left element and the superscript of the right element resulting in

$$\mathbf{r}_b^a \mathbf{r}_c^b = \mathbf{r}_c^a. \quad (8.1.14)$$

Also, with properties (8.1.13) and (8.1.14) we find

$$\mathbf{r}_b^a \mathbf{r}_a^b = \mathbf{r}_a^a = 1 \iff \mathbf{r}_b^a = (\mathbf{r}_a^b)^* \iff (\mathbf{r}_b^a)^* = \mathbf{r}_a^b, \quad (8.1.15)$$

or, simply, that the rotation that takes the object from \mathcal{F}_a to \mathcal{F}_b is the conjugate of that which takes it from \mathcal{F}_b to \mathcal{F}_a .

8.1.5 Mapping quaternions and \mathbb{R}^4

To enhance clarity, it is helpful to explicitly demonstrate any conversions between quaternions, \mathbb{H} , and \mathbb{R}^4 . This thesis employs these conversions to derive the Jacobian matrices.

For $\mathbf{h} \in \mathbb{H}$,

$$\mathbf{v}_4(\mathbf{h}) \triangleq [h_1 \quad h_2 \quad h_3 \quad h_4]^T,$$

and for its conjugate

$$\mathbf{v}_4(\underline{\mathbf{h}}^*) \triangleq \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}}_{C_4} \mathbf{v}_4(\mathbf{h}). \quad (8.1.16)$$

As, in general, quaternion multiplication is not commutative, Hamilton operators^{*2} [42, Def. 2.1.6] aids us in finding some sort of commutativity. Given $\mathbf{h}_1, \mathbf{h}_2 \in \mathbb{H}$, the Hamilton operators are matrices that satisfy

$$\mathbf{v}_4((\mathbf{h}\mathbf{g})) = \overset{+}{\mathbf{H}}_4(\mathbf{h}) \mathbf{v}_4(\mathbf{g}) = \overset{-}{\mathbf{H}}_4(\mathbf{g}) \mathbf{v}_4(\mathbf{h}).$$

Explicitly

$$\overset{+}{\mathbf{H}}_4(\mathbf{h}) \triangleq \begin{bmatrix} h_1 & -h_2 & -h_3 & -h_4 \\ h_2 & h_1 & -h_4 & h_3 \\ h_3 & h_4 & h_1 & -h_2 \\ h_4 & -h_3 & h_2 & h_1 \end{bmatrix} \quad (8.1.17)$$

and

$$\overset{-}{\mathbf{H}}_4(\mathbf{g}) \triangleq \begin{bmatrix} g_1 & -g_2 & -g_3 & -g_4 \\ g_2 & g_1 & g_4 & -g_3 \\ g_3 & -g_4 & g_1 & g_2 \\ g_4 & g_3 & -g_2 & g_1 \end{bmatrix}. \quad (8.1.18)$$

The cross product and the inner product in quaternions also have matricial analogues. From (8.1.10), one can find by direct calculation that the inner product of any two pure quaternions

^{*2}The term Hamilton operator is not commonly used.

$\mathbf{a} = \hat{i}a_2 + \hat{j}a_3 + \hat{k}a_4$ and $\mathbf{b} = \hat{i}b_2 + \hat{j}b_3 + \hat{k}b_4$ is a real number given by

$$\langle \mathbf{a}, \mathbf{b} \rangle = -\frac{\mathbf{ab} + \mathbf{ba}}{2} = \mathbf{v}_4(\mathbf{a})^T \mathbf{v}_4(\mathbf{b}) = \mathbf{v}_4(\mathbf{b})^T \mathbf{v}_4(\mathbf{a}). \quad (8.1.19)$$

The cross product between \mathbf{a} and \mathbf{b} is mapped into \mathbb{R}^4 as

$$\mathbf{v}_4((\mathbf{a} \times \mathbf{b})) = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -a_4 & a_3 \\ 0 & a_4 & 0 & -a_2 \\ 0 & -a_3 & a_2 & 0 \end{bmatrix}}_{\bar{S}(\mathbf{a})} \mathbf{v}_4(\mathbf{b}) = \bar{S}(\mathbf{a}) \mathbf{v}_4(\mathbf{b}) = \bar{S}(\mathbf{b})^T \mathbf{v}_4(\mathbf{a}). \quad (8.1.20)$$

Moreover, the time derivative of a pure quaternion is given by

$$\begin{aligned} \frac{d(\|\mathbf{a}\|)}{dt} &= \frac{\dot{\mathbf{a}}\mathbf{a}^* + \mathbf{a}\dot{\mathbf{a}}^*}{2\|\mathbf{a}\|} \\ &= \frac{\langle \mathbf{a}, \dot{\mathbf{a}} \rangle}{\|\mathbf{a}\|} \\ &= \frac{1}{\|\mathbf{a}\|} \mathbf{v}_4(\mathbf{a})^T \mathbf{v}_4(\dot{\mathbf{a}}). \end{aligned} \quad (8.1.21)$$

8.1.6 The dual quaternion set, \mathcal{H}

Dual quaternions can be seen as a number in which primary and dual parts are quaternions. They are formally defined as

$$\mathcal{H} \triangleq \left\{ \underline{\mathbf{h}} = \mathbf{h} + \varepsilon \mathbf{h}' : \mathbf{h}, \mathbf{h}' \in \mathbb{H}, \varepsilon^2 = 0, \varepsilon \neq 0 \right\}.$$

As aforementioned, $\mathbb{H} \subset \mathcal{H}$ and $\mathbb{D} \subset \mathcal{H}$. A quaternion is a dual quaternion with null dual part. A dual number is a dual quaternion with null imaginary part.

The dual quaternion set borrows the operators to retrieve the dual

$$\mathcal{D}(\underline{\mathbf{h}}) = \mathbf{h}'$$

and primary

$$\mathcal{P}(\underline{\mathbf{h}}) = \mathbf{h}$$

parts of the dual quaternion. Given $\underline{\mathbf{h}}, \underline{\mathbf{g}} \in \mathcal{H}$ those operators facilitate the definition of the sum or subtraction

$$\underline{\mathbf{h}} \pm \underline{\mathbf{g}} = \mathcal{P}(\underline{\mathbf{h}}) \pm \mathcal{P}(\underline{\mathbf{g}}) + \varepsilon (\mathcal{D}(\underline{\mathbf{h}}) + \mathcal{D}(\underline{\mathbf{g}}))$$

and the dual quaternion multiplication

$$\begin{aligned}\underline{\mathbf{h}}\underline{\mathbf{g}} &= (\mathcal{P}(\underline{\mathbf{h}}) + \varepsilon \mathcal{D}(\underline{\mathbf{h}}))(\mathcal{P}(\underline{\mathbf{g}}) + \varepsilon \mathcal{D}(\underline{\mathbf{g}})) \\ &= \mathcal{P}(\underline{\mathbf{h}})\mathcal{P}(\underline{\mathbf{g}}) + \varepsilon (\mathcal{P}(\underline{\mathbf{h}})\mathcal{D}(\underline{\mathbf{g}}) + \mathcal{D}(\underline{\mathbf{h}})\mathcal{P}(\underline{\mathbf{g}})).\end{aligned}$$

Also, the operators to retrieve the real part

$$\begin{aligned}\text{Re}(\underline{\mathbf{h}}) &= \text{Re}(\mathbf{h}) + \varepsilon \text{Re}(\mathbf{h}') \\ &= h_1 + \varepsilon h'_1,\end{aligned}$$

and imaginary parts

$$\begin{aligned}\text{Im}(\underline{\mathbf{h}}) &= \text{Im}(\mathbf{h}) + \varepsilon \text{Im}(\mathbf{h}') \\ &= (\hat{i}h_2 + \hat{j}h_3 + \hat{k}h_4) + \varepsilon (\hat{i}h'_2 + \hat{j}h'_3 + \hat{k}h'_4).\end{aligned}$$

The conjugate of a dual quaternion can be defined as

$$\underline{\mathbf{h}}^* \triangleq \text{Re}(\underline{\mathbf{h}}) - \text{Im}(\underline{\mathbf{h}}) \quad (8.1.22)$$

and norm is defined as

$$\|\underline{\mathbf{h}}\| \triangleq \sqrt{\underline{\mathbf{h}}\underline{\mathbf{h}}^*} = \sqrt{\underline{\mathbf{h}}^*\underline{\mathbf{h}}}. \quad (8.1.23)$$

8.1.7 The pure dual quaternion set, \mathcal{H}_p

The pure dual quaternion set is defined as

$$\mathcal{H}_p \triangleq \left\{ \underline{\mathbf{h}} \in \mathcal{H} : \text{Re}(\underline{\mathbf{h}}) = 0 \right\},$$

and the inner product

$$\langle \underline{\mathbf{h}}, \underline{\mathbf{g}} \rangle = \langle \underline{\mathbf{g}}, \underline{\mathbf{h}} \rangle \triangleq -\frac{\underline{\mathbf{h}}\underline{\mathbf{g}} + \underline{\mathbf{g}}\underline{\mathbf{h}}}{2} \quad (8.1.24)$$

and the cross product

$$\underline{\mathbf{h}} \times \underline{\mathbf{g}} = -\underline{\mathbf{h}} \times \underline{\mathbf{g}} \triangleq \frac{\underline{\mathbf{h}}\underline{\mathbf{g}} - \underline{\mathbf{g}}\underline{\mathbf{h}}}{2}. \quad (8.1.25)$$

8.1.8 The unit dual quaternion set, $\underline{\mathcal{S}}$, and rigid body motion

The unit dual quaternion set is comprised of dual quaternions with unit norm, see (8.1.23), and defined as

$$\underline{\mathcal{S}} \triangleq \left\{ \underline{\mathbf{h}} \in \mathcal{H} : \|\underline{\mathbf{h}}\| = 1 \right\},$$

and can be used to describe poses, which are combined translations and rotations. Considering elements $\mathbf{r} \in \mathbb{S}^3$ and $\mathbf{t} \in \mathbb{H}_p$, the dual quaternion representing the translation \mathbf{t} followed by a rotation \mathbf{r} is given by

$$\underline{\mathbf{x}} = \mathbf{r} + \frac{1}{2}\varepsilon \mathbf{t} \mathbf{r},$$

which is always unit norm since

$$\begin{aligned} \underline{\mathbf{x}} \underline{\mathbf{x}}^* &= \left(\mathbf{r} + \frac{1}{2}\varepsilon \mathbf{t} \mathbf{r} \right) \left(\mathbf{r} + \frac{1}{2}\varepsilon \mathbf{t} \mathbf{r} \right)^* \\ &= \left(\mathbf{r} + \frac{1}{2}\varepsilon \mathbf{t} \mathbf{r} \right) \left(\mathbf{r}^* + \frac{1}{2}\varepsilon \mathbf{r}^* \mathbf{t}^* \right) \\ &\stackrel{(8.1.7)}{=} \left(\mathbf{r} + \frac{1}{2}\varepsilon \mathbf{t} \mathbf{r} \right) \left(\mathbf{r}^* - \frac{1}{2}\varepsilon \mathbf{r}^* \mathbf{t} \right) \\ &\stackrel{(8.1.12)}{=} \mathbf{r} \mathbf{r}^* + \frac{1}{2}\varepsilon (\mathbf{r} \mathbf{r}^* \mathbf{t} - \mathbf{t} \mathbf{r} \mathbf{r}^*) \\ &= 1. \end{aligned}$$

The translation of a unit dual quaternion $\underline{\mathbf{x}} \in \underline{\mathcal{S}}$ can be retrieved by the translation operator

$$\begin{aligned} \mathcal{T}(\underline{\mathbf{x}}) &= 2 \mathcal{D}(\underline{\mathbf{x}}) \mathcal{P}(\underline{\mathbf{x}}^*) \\ &= 2 \left(\frac{1}{2} \mathbf{t} \mathbf{r} \right) (\mathbf{r}^*) \\ &= \mathbf{t}. \end{aligned}$$

A pose of a frame \mathcal{F}_a with respect to itself is always

$$\underline{\mathbf{x}}_a^a = 1.$$

To move from a frame \mathcal{F}_a to a frame \mathcal{F}_b , we post-multiply by the relative pose

$$\underline{\mathbf{x}}_a^a \underline{\mathbf{x}}_b^a = \underline{\mathbf{x}}_b^a. \quad (8.1.26)$$

The opposite relative pose is given by the conjugate

$$\underline{x}_b^a \underline{x}_a^b = \underline{x}_a^a = 1 \iff \underline{x}_b^a = (\underline{x}_a^b)^* \iff (\underline{x}_b^a)^* = \underline{x}_a^b.$$

8.1.9 Mapping dual quaternions and \mathbb{R}^8

Similarly to quaternions, any conversions between \mathcal{H} and \mathbb{R}^8 will be explicit.

For $\underline{h} \in \mathcal{H}$,

$$v_8(\underline{h}) \triangleq [h_1 \ h_2 \ h_3 \ h_4 \ h'_1 \ h'_2 \ h'_3 \ h'_4]^T,$$

and for its conjugate

$$v_8(\underline{h}^*) \triangleq \begin{bmatrix} C_4 & 0 \\ 0 & C_4 \end{bmatrix} v_8(\underline{h}). \quad (8.1.27)$$

Similarly to quaternions, for $\underline{h}, \underline{g} \in \mathcal{H}$, the Hamilton operators for dual quaternions are matrices that satisfy

$$v_8(\underline{h}\underline{g}) = \dot{\underline{H}}_8(\underline{h}) v_8(\underline{g}) = \bar{\underline{H}}_8(\underline{g}) v_8(\underline{h}).$$

Explicitly

$$\dot{\underline{H}}_8(\underline{h}) \triangleq \begin{bmatrix} \dot{\underline{H}}_4(\mathcal{P}(\underline{h})) & 0 \\ \dot{\underline{H}}_4(\mathcal{D}(\underline{h})) & \dot{\underline{H}}_4(\mathcal{P}(\underline{h})) \end{bmatrix}. \quad (8.1.28)$$

and

$$\bar{\underline{H}}_8(\underline{g}) \triangleq \begin{bmatrix} \bar{\underline{H}}_4(\mathcal{P}(\underline{g})) & 0 \\ \bar{\underline{H}}_4(\mathcal{D}(\underline{g})) & \bar{\underline{H}}_4(\mathcal{P}(\underline{g})) \end{bmatrix}. \quad (8.1.29)$$

8.2 Robot kinematics using unit dual quaternions

Mathematically, we begin by defining $\underline{q} \triangleq \underline{q}(t) \in \mathbb{R}^n$

$$\underline{q} = [q_1 \ q_2 \ \dots \ q_n]^T$$

as the vector of n *joint space* values. These values can either be meters for prismatic joints or radians for rotational joints. The objective of robot kinematics is to map the joint space to the *task space*. The task space is a vector space composed by usually linear independent variables that make sense to a given task. Formally, we define the vector of m task space variables as

$$\underline{x} \triangleq \underline{x}(\underline{q}) \in \mathbb{R}^m$$

$$\underline{x} = [x_1 \quad x_2 \quad \dots \quad x_m]^T.$$

8.2.1 Forward kinematics model

A first requirement in any method of control is the *forward kinematics model* (FKM). It is a function that maps joint space values into task space values. Formally, the FKM is the function $f(\underline{q})$ such that

$$f(\underline{q}(t)) = \underline{x}(\underline{q}), \quad (8.2.1)$$

which means that in any given *joint configuration*, \underline{q} , we can map joint space values into task space values. The task space values are commonly associated with the tip of an end effector attached to the last link of the robot, but it can refer to any other point somehow related to the joint motions.

The FKM in (8.2.1) is generic and depends on the task space values in question. We discuss, initially, in terms of the pose. Translation and rotation can be obtained from the pose FKM.

8.2.2 Forward kinematics models for pose, translation, and rotation

Supposing several robots and other elements can share the same space, first we define a world frame, \mathcal{F}_w . Without loss of generality, we pick one robot in this world and give it a base frame, \mathcal{F}_0 . The constant transformation between those is given by

$$\underline{x}_0^w \in \underline{S}.$$

The forward kinematics model depends on how the robot is initially modeled. In general, if we label the joints with reference frame \mathcal{F}_i using $i = 0 \dots n - 1$, the transformation of each joint from the current to the next is

$$\underline{x}_{i+1}^i(q_i) \triangleq \underline{x}_{i+1}^i \in \underline{S}, \quad (8.2.2)$$

each of which depends only on the current joint value. Lastly, we define the constant transformation between the last joint and the frame of a point \mathcal{F}_p , such as the end effector tip, as

$$\underline{x}_p^n \in \underline{S}.$$

It is important to have explicitly defined \underline{x}_0^w and \underline{x}_p^n since they will be required when we discuss on calibration.

With those definitions in mind, the FKM, in this case, is to find

$$\underline{x}_p^w(\boldsymbol{q}) \in \underline{S},$$

which is the transformation from \mathcal{F}_w to \mathcal{F}_p .

The FKM is found as the sequential post-multiplication of (8.2.2) for each joint, as

$$f(\boldsymbol{q}(t)) \triangleq \underline{x}_p^w(\boldsymbol{q}) \triangleq \underline{x}_0^w \underline{x}_1^0(q_0) \underline{x}_2^1(q_1) \dots \underline{x}_{n-1}^{n-1}(q_{n-1}) \underline{x}_p^n. \quad (8.2.3)$$

The translation can be obtained as

$$\boldsymbol{t}_p^w(\boldsymbol{q}) = \mathcal{T}(\underline{x}_p^w(\boldsymbol{q})) \quad (8.2.4)$$

and the rotation as

$$\boldsymbol{r}_p^w(\boldsymbol{q}) = \mathcal{P}(\underline{x}_p^w(\boldsymbol{q})). \quad (8.2.5)$$

What remains is to describe each joint in the form (8.2.2). In this section, two methods for modeling robot kinematics are shown, due to the robotic systems used in the following chapters: the classic Denavit-Hartenberg parameters in Section 8.2.3 and DENSO Corporations' proprietary parameters in Section 8.2.4. In this thesis, the DENSO parameters are primarily used for the higher accuracy calibration.

8.2.3 Denavit-Hartenberg parameters

The Denavit-Hartenberg (DH) parameters are arguably the most well-known kinematic parameterization of robots. It is based on breaking down the transformation between joints as the four sequential transformations, namely

1. rotation about the z -axis of \mathcal{F}_i by θ_i taking it to $\mathcal{F}_{i'}$

$$\underline{x}_{i'}^i(\theta_i) \triangleq \cos \frac{\theta_i}{2} + \hat{k} \sin \frac{\theta_i}{2}, \quad (8.2.6)$$

2. translation in the z -axis of $\mathcal{F}_{i'}$ by d_i taking it to $\mathcal{F}_{i''}$

$$\underline{x}_{i''}^{i'}(d_i) \triangleq 1 + \frac{1}{2}\varepsilon d_i \hat{k}, \quad (8.2.7)$$

3. translation in the x -axis of $\mathcal{F}_{i''}$ by a_i taking it to $\mathcal{F}_{i'''}$

$$\underline{x}_{i'''}^{i''}(a_i) \triangleq 1 + \frac{1}{2}\varepsilon a_i \hat{i},$$

4. rotation about the x -axis of $\mathcal{F}_{i'''}$ by α_i taking it to \mathcal{F}_{i+1}

$$\underline{x}_{i+1}^{i'''}(\alpha_i) \triangleq \cos \frac{\alpha_i}{2} + \hat{i} \sin \frac{\alpha_i}{2}.$$

The final transformation can be found by the sequential post-multiplication

$$\underline{x}_{i+1}^i = \underline{x}_{i'}^i(\theta_i) \underline{x}_{i''}^{i'}(d_i) \underline{x}_{i'''}^{i''}(a_i) \underline{x}_{i+1}^{i'''}(\alpha_i). \quad (8.2.8)$$

One may choose to further simplify (8.2.8), but that is not required. Even in a computational implementation, leaving (8.2.8) in that form eases readability with little loss of performance.

This parameterization expects the actuation to be done either in (8.2.6) for a rotational joint or in (8.2.7) for a prismatic joint. For a rotational joint, (8.2.8) becomes

$$\underline{x}_{i+1}^i = \underline{x}_{i'}^i(\theta_i + q_i) \underline{x}_{i''}^{i'}(d_i) \underline{x}_{i'''}^{i''}(a_i) \underline{x}_{i+1}^{i'''}(\alpha_i), \quad (8.2.9)$$

and for a prismatic joint

$$\underline{x}_{i+1}^i = \underline{x}_{i'}^i(\theta_i) \underline{x}_{i''}^{i'}(d_i + q_i) \underline{x}_{i'''}^{i''}(a_i) \underline{x}_{i+1}^{i'''}(\alpha_i). \quad (8.2.10)$$

8.2.4 DENSO Corporations' proprietary parameters

DENSO Corporation holds a patent of what they call high accuracy calibration parameters. It is also composed of four sequential transformations, but with six parameters. Namely,

1. rotation about the z -axis of \mathcal{F}_i by γ_i taking it to $\mathcal{F}_{i'}$

$$\underline{\mathbf{x}}_{i'}^i(\gamma_i) \triangleq \cos \frac{\gamma_i}{2} + \hat{k} \sin \frac{\gamma_i}{2},$$

2. combined translation of $\mathcal{F}_{i'}$ x -axis by a_i , $\mathcal{F}_{i'}$ y -axis by b_i , and $\mathcal{F}_{i'}$ z -axis by d_i , taking it to $\mathcal{F}_{i''}$

$$\underline{\mathbf{x}}_{i''}^{i'}(a_i, b_i, d_i) \triangleq 1 + \frac{1}{2}\varepsilon(a_i\hat{i} + b_i\hat{j} + d_i\hat{k}),$$

3. rotation about the x -axis of $\mathcal{F}_{i''}$ by α_i taking it to $\mathcal{F}_{i'''}$

$$\underline{\mathbf{x}}_{i'''}^{i''}(\alpha_i) \triangleq \cos \frac{\alpha_i}{2} + \hat{i} \sin \frac{\alpha_i}{2},$$

4. rotation about the y -axis of $\mathcal{F}_{i'''}$ by β_i taking it to \mathcal{F}_{i+1}

$$\underline{\mathbf{x}}_{i+1}^{i'''}(\beta_i) \triangleq \cos \frac{\beta_i}{2} + \hat{j} \sin \frac{\beta_i}{2}.$$

These parameters are only used for rotational joints, therefore the final transformation is given by

$$\underline{\mathbf{x}}_{i+1}^i = \underline{\mathbf{x}}_{i'}^i(\gamma_i + q_i) \underline{\mathbf{x}}_{i''}^{i'}(a_i, b_i, d_i) \underline{\mathbf{x}}_{i'''}^{i''}(\alpha) \underline{\mathbf{x}}_{i+1}^{i'''}(\beta). \quad (8.2.11)$$

8.2.5 Inverse kinematics

After finding the FKM, $f(\mathbf{q}(t))$, of a manipulator, controlling the robot requires finding some sort of *inverse kinematics* function

$$f^{-1}(\mathbf{x}(\mathbf{q})) = \mathbf{q}(t). \quad (8.2.12)$$

The FKM gives us, for instance, *the tool pose given the joint values*, while the inverse kinematics tells us *how the joint values need to be for a given tool pose*. A quick inspection of

a robotics textbook will show how painstaking is the process of finding the closed form inverse kinematics (8.2.12) of even the simplest types of manipulators. In many cases, those closed form functions are impossible to find unless additional constraints are imposed due to infinite solutions which may appear when *joint space degree-of-freedom (n) > task space degree-of-freedom (m)*.

8.2.6 Differential Kinematics

It is common practice, instead, to initially find a linear relationship between *joint space velocities*, $\dot{\mathbf{q}}$, and *task space velocities*, $\dot{\mathbf{x}}$, as

$$\dot{\mathbf{x}} \underset{8.2.1}{=} \frac{\partial(\mathbf{x}(\mathbf{q}))}{\partial t} = \underbrace{\frac{\partial(f(\mathbf{q}(t)))}{\partial \mathbf{q}}}_{\mathbf{J}} \frac{\partial(\mathbf{q}(t))}{\partial t} = \mathbf{J}\dot{\mathbf{q}}, \quad (8.2.13)$$

in which $\mathbf{J} \in \mathbb{R}^{m \times n}$ a Jacobian, a commonly used concept in matrix algebra. Notice that any dependency on second or higher order derivatives of \mathbf{x} are not present in (8.2.13), which means it is only precise when low accelerations are imposed. In case of a unit dual quaternion FKM $\underline{\mathbf{x}} \triangleq \underline{\mathbf{x}}^w(\mathbf{q})$ such as (8.2.3), (notice that $\underline{\mathbf{x}}$ is different from \mathbf{x} , $\underline{\mathbf{x}}$ is a unit dual quaternion in task space, whereas \mathbf{x} is a vector in task space. We use an operator $v_8()$ to convert dual quaternions to vectors. For $\underline{\mathbf{h}} \in \mathcal{H}$, $v_8(\underline{\mathbf{h}}) \triangleq [h_1 \ h_2 \ h_3 \ h_4 \ h'_1 \ h'_2 \ h'_3 \ h'_4]^T$) the differential kinematics becomes

$$v_8(\underline{\dot{\mathbf{x}}}) = \underbrace{\frac{\partial(v_8(\underline{\mathbf{x}}(\mathbf{q})))}{\partial \mathbf{q}}}_{\mathbf{J}_{\underline{\mathbf{x}}}} \dot{\mathbf{q}}, \quad (8.2.14)$$

in which $\mathbf{J}_{\underline{\mathbf{x}}} \in \mathbb{R}^{8 \times n}$ is the unit dual quaternion pose Jacobian. The rotation and translation Jacobians are easily obtained from $\mathbf{J}_{\underline{\mathbf{x}}}$. To do so, we first partition (8.2.14) into its primary and dual parts

$$\begin{bmatrix} v_4(\mathcal{P}(\underline{\dot{\mathbf{x}}})) \\ v_4(\mathcal{D}(\underline{\dot{\mathbf{x}}})) \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{\mathcal{P}(\underline{\mathbf{x}})} \\ \mathbf{J}_{\mathcal{D}(\underline{\mathbf{x}})} \end{bmatrix} \dot{\mathbf{q}}.$$

The rotation Jacobian $\mathbf{J}_r \in \mathbb{R}^{4 \times n}$ is found as

$$\begin{aligned} \mathbf{r} &= \mathcal{P}(\underline{x}) \\ \iff \frac{d}{dt}(\mathbf{r}) &= \frac{d}{dt}(\mathcal{P}(\underline{x})) \\ \iff \mathbf{v}_4(\dot{\mathbf{r}}) &= \underbrace{\mathbf{J}_{\mathcal{P}(\underline{x})}}_{\mathbf{J}_r} \dot{\mathbf{q}}. \end{aligned} \quad (8.2.15)$$

The translation Jacobian $\mathbf{J}_t \in \mathbb{R}^{4 \times n}$ can be found with a bit of quaternion algebra (see 8.1.16, 8.1.17, and 8.1.18)

$$\begin{aligned} \mathcal{T}(\underline{x}) &= \mathbf{t} = 2 \mathcal{D}(\underline{x}) (\mathcal{P}(\underline{x}))^* \\ \iff \frac{d}{dt}(\mathbf{t}) &= 2 \frac{d}{dt}(\mathcal{D}(\underline{x})) (\mathcal{P}(\underline{x}))^* + 2 \mathcal{D}(\underline{x}) \frac{d}{dt}((\mathcal{P}(\underline{x}))^*) \\ \iff \mathbf{v}_4(\dot{\mathbf{t}}) &= 2 \underbrace{\left(\bar{\mathbf{H}}_4((\mathcal{P}(\underline{x}))^*) \mathbf{J}_{\mathcal{D}(\underline{x})} + \dot{\mathbf{H}}_4(\mathcal{D}(\underline{x})) \mathbf{C}_4 \mathbf{J}_{\mathcal{P}(\underline{x})} \right)}_{\mathbf{J}_t} \dot{\mathbf{q}}. \end{aligned} \quad (8.2.16)$$

8.2.7 The unit dual quaternion Jacobian, $\mathbf{J}_{\underline{x}}$

Finding $\mathbf{J}_{\underline{x}}$ depends on the FKM we use. We start with the generic FKM (8.2.3)

$$\begin{aligned} f(\mathbf{q}(t)) &\triangleq \underline{x}_p^w(\mathbf{q}) = \underline{x}_0^w \underline{x}_1^0(q_0) \underline{x}_2^1(q_1) \dots \underline{x}_n^{n-1}(q_{n-1}) \underline{x}_p^n \\ &= \underline{x}_0^w \prod_{i=0}^{n-1} \underline{x}_{i+1}^i(q_i) \underline{x}_p^n. \end{aligned} \quad (8.2.17)$$

Notice that the partial derivative of (8.2.17) with respect to any given joint value q_j , $j = 0 \dots n - 1$, is obtained using the chain rule

$$\begin{aligned} \frac{\partial(\underline{x}_p^w(\mathbf{q}))}{\partial q_j} &= \frac{\partial(\underline{x}_0^w \prod_{i=0}^{n-1} \underline{x}_{i+1}^i(q_i) \underline{x}_p^n)}{\partial q_j} \\ &= \underline{x}_0^w \prod_{i=0}^{j-1} (\underline{x}_{i+1}^i(q_i)) \frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j} \prod_{i=j+1}^{n-1} (\underline{x}_{i+1}^i(q_i)) \underline{x}_p^n \\ &= \underline{x}_{j-1}^w \frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j} \underline{x}_p^{j+1}. \end{aligned}$$

Therefore, to find the Jacobian for a given FKM, we only need to find $\frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j}$ for each joint j and compose them into the $m \times n$ matrix

$$\mathbf{J}_{\underline{x}} = \left[v_8 \left(\underline{x}_0^w \frac{\partial(\underline{x}_1^0(q_0))}{\partial q_0} \underline{x}_p^1 \right) \quad v_8 \left(\underline{x}_1^w \frac{\partial(\underline{x}_2^1(q_1))}{\partial q_1} \underline{x}_p^2 \right) \quad \dots \quad v_8 \left(\underline{x}_{n-1}^w \frac{\partial(\underline{x}_n^{n-1}(q_{n-1}))}{\partial q_{n-1}} \underline{x}_p^n \right) \right].$$

In the following, we will see how to find $\frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j}$ for each joint j for robots modeled using the standard Denavit-Hartenberg parameters and the the DENSO Corporation's proprietary parameters.

Jacobian for robots modeled using the standard Denavit-Hartenberg parameters

For a rotational joint, we find (see 8.2.9)

$$\begin{aligned} \frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j} &= \frac{\partial(\underline{x}_{j'}^j(\theta_j + q_j))}{\partial q_j} \underline{x}_{j''}^{j'}(d_j) \underline{x}_{j'''}^{j''}(a_j) \underline{x}_{j+1}^{j'''}(\alpha_j) \\ &= \left(-\sin \frac{\theta_j + q_j}{2} + \hat{k} \cos \frac{\theta_j + q_j}{2} \right) \underline{x}_{j'}^{j'}(d_j) \underline{x}_{j'''}^{j''}(a_j) \underline{x}_{j+1}^{j'''}(\alpha_j) \end{aligned}$$

and for a prismatic joint (see 8.2.10)

$$\begin{aligned} \frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j} &= \underline{x}_{j'}^j(\theta_j) \frac{\partial(\underline{x}_{j''}^{j'}(d_j + q_j))}{\partial q_j} \underline{x}_{j'''}^{j''}(a_j) \underline{x}_{j+1}^{j'''}(\alpha_j) \\ &= \underline{x}_{j'}^j(\theta_j) \left(\frac{1}{2} \varepsilon \hat{k} \right) \underline{x}_{j'''}^{j''}(a_j) \underline{x}_{j+1}^{j'''}(\alpha_j). \end{aligned}$$

Jacobians for robots modeled using the DENSO Corporation proprietary parameters.

For the rotational joint, we find (see 8.2.11)

$$\frac{\partial(\underline{x}_{j+1}^j(q_j))}{\partial q_j} = \left(-\sin \frac{\gamma_j + q_j}{2} + \hat{k} \cos \frac{\gamma_j + q_j}{2} \right) \underline{x}_{j''}^{j'}(a_j, b_j, d_j) \underline{x}_{j'''}^{j''}(a) \underline{x}_{j+1}^{j'''}(\beta).$$

8.3 Constrained optimization algorithm

In this thesis, a centralized kinematic control strategy proposed in [39] is used to control the two robotic arms of the SmartArm surgical robotic system.

Let $\mathbf{t}_i, \mathbf{t}_{i,d} \in \mathbb{H}_p$ be the translation and the desired translation of the end effector of each i -th robot, with $i \in \{1, 2\}$. Moreover, let $\mathbf{r}_i, \mathbf{r}_{i,d} \in \mathbb{S}^3$ be the rotation and the desired rotation of the end effector of each i -th robot. Given $\mathbf{q} = [\mathbf{q}_1^T \quad \mathbf{q}_2^T]^T$, the desired control signal, $\mathbf{u} = [\mathbf{u}_1^T \quad \mathbf{u}_2^T]^T$, is obtained as

$$\begin{aligned} \mathbf{u} &\in \underset{\dot{\mathbf{q}}}{\operatorname{argmin}} \beta \mathcal{F}_1 + (1 - \beta) \mathcal{F}_2 \\ \text{subject to } \mathbf{W}\dot{\mathbf{q}} &\leq \mathbf{w}, \end{aligned} \tag{8.3.1}$$

where

$$\mathcal{F}_i = \alpha f_{t,i} + (1 - \alpha) f_{r,i} + f_{\lambda,i} \tag{8.3.2}$$

in which $f_{t,i} \triangleq \|\mathbf{J}_{\mathbf{t}_i} \dot{\mathbf{q}}_i + \eta \mathbf{v}_4(\tilde{\mathbf{t}}_i)\|_2^2$, $f_{r,i} \triangleq \|\mathbf{J}_{\mathbf{r}_i} \dot{\mathbf{q}}_i + \eta \mathbf{v}_4(\tilde{\mathbf{r}}_i)\|_2^2$, and $f_{\lambda,i} \triangleq \lambda \|\dot{\mathbf{q}}_i\|_2^2$, are the cost functions related to the end-effector translation, end-effector rotation, and joint velocities of

the i -th robot. Furthermore, each robot has a translation Jacobian $\mathbf{J}_{t_i} \in \mathbb{R}^{4 \times n_i}$ that satisfies $\mathbf{v}_4(\dot{\mathbf{t}}_i) = \mathbf{J}_{t_i}\dot{\mathbf{q}}_i$ (see (8.2.16)), a rotation Jacobian $\mathbf{J}_{r_i} \in \mathbb{R}^{4 \times n_i}$ that satisfies $\mathbf{v}_4(\dot{\mathbf{r}}_i) = \mathbf{J}_{r_i}\dot{\mathbf{q}}_i$ (see 8.2.15), a translation error $\tilde{\mathbf{t}}_i \triangleq \tilde{\mathbf{t}}_i(\mathbf{q}_i) = \mathbf{t}_i - \mathbf{t}_{i,d}$, and a switching rotational error

$$\tilde{\mathbf{r}}_i \triangleq \begin{cases} (\mathbf{r}_i)^* \mathbf{r}_{i,d} - 1 & \text{if } \|\mathbf{r}_i^* \mathbf{r}_{i,d} - 1\|_2 < \|\mathbf{r}_i^* \mathbf{r}_{i,d} + 1\|_2 \\ (\mathbf{r}_i)^* \mathbf{r}_{i,d} + 1 & \text{otherwise.} \end{cases}$$

In addition, $\eta \in (0, \infty) \subset \mathbb{R}$ is a tunable gain, $\lambda \in [0, \infty) \subset \mathbb{R}$ is a damping factor, and $\alpha, \beta \in [0, 1] \subset \mathbb{R}$ are weights used to define the priorities between the translation and the rotation and between robots respectively. Also, $\dot{\mathbf{q}} = [\dot{\mathbf{q}}_1^T \quad \dot{\mathbf{q}}_2^T]^T$ and $\Lambda \in \mathbb{R}^{n \times n}$ are a positive definite damping matrix, usually diagonal. Since the values of these parameters vary depending on the system and the task, they are adjusted by running simulations in advance.

The inequality constraints $\mathbf{W}\dot{\mathbf{q}} \leq \mathbf{w}$, in which $\mathbf{W} \triangleq \mathbf{W}(\mathbf{q}) \in \mathbb{R}^{r \times (n_1+n_2)}$, $\mathbf{w} \triangleq \mathbf{w}(\mathbf{q}) \in \mathbb{R}^r$ are used to generate active constraints using the vector-field inequalities (VFI) method [36]. Problem (8.3.1) in general does not have an analytical solution and a numerical solver must be used. The Lyapunov stability of such approaches has been proven in [45], as long as the objective function is convex and $\mathbf{u} = 0$ is in the feasible set.

The switching unit quaternion controller

Since the group of unit quaternions double covers $\text{SO}(3)$, both $\mathbf{r} \in \mathbb{S}^3$ and $-\mathbf{r}$ represent the same orientation, which leads to the unwinding problem [46]. In practice, this problem results in undesired movements. For example, if the orientation error is defined as $(\mathbf{r}_i)^* \mathbf{r}_{i,d} - 1$, the orientation error is zero when $\mathbf{r}_{i,d} = \mathbf{r}_i$. However, if $\mathbf{r}_{i,d} = -\mathbf{r}_i$, the orientation error becomes -2 even though the current orientation is already correct. This causes the robot moves unnecessarily until it reaches a new equilibrium point. To circumvent this problem, discontinuous or hybrid control laws can be employed [46]. In the case discussed in [39], the problem is achieved by switching the error: If $(\mathbf{r}_i)^* \mathbf{r}_{i,d}$ is closer to 1 , the error is given by $(\mathbf{r}_i)^* \mathbf{r}_{i,d} - 1$; conversely, if $(\mathbf{r}_i)^* \mathbf{r}_{i,d}$ is closer to -1 , the error is given by $(\mathbf{r}_i)^* \mathbf{r}_{i,d} + 1$.

The translation and rotation weight, α

The weight $\alpha \in [0, 1]$ is used to balance translational and rotational gains. In the vitreoretinal setup, the instruments are straight and possess only one degree of freedom around their shafts. While the motions of the instruments are fixed at the insertion points, however, a 7th rotational joint has been added to the dominant surgical instrument. This addition allows the forceps to rotate independently of the rest of the robot, eliminating the need for transnational movement. Consequently, in this thesis, the rotational weight α is incorporated into the control law for the dominant instrument tip, whereas it is not applied to the light guide.

The robot prioritization weight, β

Although the joint velocities of the two robots are calculated simultaneously, the weight $\beta \in [0, 1]$ allows one robot to be prioritized over another. This establishes a “soft” priority between the two robots. A “soft” priority means that there is a relative level of importance between the robots, defined by β , offering a flexible and customizable approach rather than a rigid hierarchical scheme.

8.4 Vector-field-inequalities method

The vector-field-inequalities (VFI) method [36] is used to map task-space constraints into configuration-space constraints. The VFI method has been shown to be particularly useful for dynamic-active constraints [47, Section III].

The VFI method relies on obtaining functions $d \triangleq d(\mathbf{q}, t) \in \mathbb{R}$ that represent the signed

distance between two geometric primitives. The time-derivative of the distance is

$$\dot{d} = \underbrace{\frac{\partial(d(\mathbf{q}, t))}{\partial \mathbf{q}}}_{\mathbf{J}_d} \dot{\mathbf{q}} + \zeta(t),$$

where $\mathbf{J}_d \in \mathbb{R}^{1 \times n}$ is the distance Jacobian and $\zeta(t) = \dot{d}$ is the residual, that contains the distance dynamics unrelated to $\dot{\mathbf{q}}$. Moreover, let there be a safe distance $d_{\text{safe}} \triangleq d_{\text{safe}}(t) \in [0, \infty)$ for safe and restricted zones.

Using these definitions, and given $\eta_d \in [0, \infty)$, the signed distance dynamics is constrained by $\dot{\tilde{d}} \geq -\eta_d \tilde{d}$ in both cases, which actively filters the robot motion only in the direction approaching the boundary between the primitives so that the primitives do not collide.

The following constraint is used to generate safe zones, such as the entry-point constraint,

$$\mathbf{J}_d \dot{\mathbf{q}} \leq \eta_d \tilde{d} - \zeta_{\text{safe}}(t), \quad (8.4.1)$$

Alternatively, restricted zones, such as constraints to prevent collisions, are generated by

$$-\mathbf{J}_d \dot{\mathbf{q}} \leq \eta_d \tilde{d} + \zeta_{\text{safe}}(t). \quad (8.4.2)$$

In order to use the VFI in (8.4.1) and (8.4.2), Marinho *et al.* [36] defined the (squared) distance functions for relevant geometrical primitives (point, line, and plane); then, they found the corresponding Jacobians and residuals. These geometrical primitives can be easily combined to obtain other primitives. For instance, a point with a positive scalar yields a sphere, whereas a line with a positive scalar yields an infinite cylinder. The intersection between an infinite cylinder and two parallel planes results in a finite cylinder. Polyhedra can be defined as intersections of planes.

This thesis uses some distance functions and corresponding Jacobians derived in [36] and are shown in Table 8.1.

Table 8.1: Summary of distance functions and corresponding Jacobians derived

Primitives	Distance function	Jacobian
Point-to-Point	$D_{t,p} \triangleq d_{t,p}^2$	$\mathbf{J}_{t,p}$
Point-to-Line	$D_{t,l} \triangleq d_{t,l}^2$	$\mathbf{J}_{t,l}$
Line-to-Point	$D_{l_z,p} \triangleq d_{l_z,p}^2$	$\mathbf{J}_{l_z,p}$
Point-to-Plane	$D_{t,\pi} \triangleq d_{t,\pi}^2$	$\mathbf{J}_{t,\pi}$

Bibliography

- [1] H. Yoo and B. J. Kim, “History and recent advances in microsurgery,” vol. 26, no. 3, pp. 174–183. [Online]. Available: <http://handmicro.org/journal/view.php?doi=10.12790/ahm.21.0097>
- [2] I. Badash, D. J. Gould, and K. M. Patel, “Supermicrosurgery: History, applications, training and the future,” vol. 5, p. 23. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5931174/>
- [3] S. Singh and C. Riviere, “Physiological tremor amplitude during retinal microsurgery,” in *Proceedings of the IEEE 28th Annual Northeast Bioengineering Conference (IEEE Cat. No.02CH37342)*, pp. 171–172. [Online]. Available: <https://ieeexplore.ieee.org/document/999520/?arnumber=999520>
- [4] P. K. Gupta, P. S. Jensen, and E. de Juan, “Surgical forces and tactile perception during retinal microsurgery,” in *Medical Image Computing and Computer-Assisted Intervention - MICCAI' 99*, C. Taylor and A. Colchester, Eds. Springer, pp. 1218–1225.
- [5] K. Faridpooya, S. H. M. van Romunde, S. S. Manning, J. C. van Meurs, G. J. L. Naus, M. J. Beelen, T. C. M. Meenink, J. Smit, and M. D. de Smet, “Randomised controlled trial on robot-assisted versus manual surgery for pucker peeling,” vol. 50, no. 9, pp. 1057–1064, _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/ceo.14174>. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/ceo.14174>
- [6] J. W. Kim, S. Wei, P. Zhang, P. Gehlbach, J. U. Kang, I. Iordachita, and M. Kobilarov, “Towards autonomous retinal microsurgery using RGB-d images,” vol. 9, no. 4, pp.

- 3807–3814, conference Name: IEEE Robotics and Automation Letters. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/10443071>
- [7] M. Esfandiari, J. W. Kim, B. Zhao, G. Amirkhani, M. Hadi, P. Gehlbach, R. H. Taylor, and I. Iordachita, “Cooperative vs. teleoperation control of the steady hand eye robot with adaptive sclera force control: A comparative study,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8209–8215. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/10611084/figures#figures>
- [8] N. Wang, X. Zhang, M. Li, H. Zhang, D. Stoyanov, and A. Stilli, “A 5-DOFs robot for posterior segment eye microsurgery,” vol. 7, no. 4, pp. 10128–10135, conference Name: IEEE Robotics and Automation Letters. [Online]. Available: <https://ieeexplore.ieee.org/document/9832732>
- [9] N. Wang, X. Zhang, D. Stoyanov, H. Zhang, and A. Stilli, “Vision-and-force-based compliance control for a posterior segment ophthalmic surgical robot,” vol. 8, no. 11, pp. 6875–6882, conference Name: IEEE Robotics and Automation Letters. [Online]. Available: <https://ieeexplore.ieee.org/document/10243078>
- [10] S. Dehghani, M. Sommersperger, M. Saleh, A. Alikhani, B. Busam, P. Gehlbach, I. Iordachita, N. Navab, and M. Ali Nasseri, “Colibri5: Real-time monocular 5-DoF trocar pose tracking for robot-assisted vitreoretinal surgery,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4547–4554. [Online]. Available: <https://ieeexplore.ieee.org/document/10610576>
- [11] Z. Li, Y. Deng, Q. Ye, W. Yu, H. Qi, Y. Liu, Z. Yu, and G.-B. Bian, “Procedure recognition by knowledge-driven segmentation in robotic-assisted vitreoretinal surgery,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10779–10785. [Online]. Available: <https://ieeexplore.ieee.org/document/10610315>
- [12] Y. Koyama, “Autonomously coordinated two-arm manipulation for robot-assisted vitreoretinal surgery,” Ph.D. dissertation, The University of Tokyo, 2023.
- [13] G. Fichtinger, P. Kazanzides, A. M. Okamura, G. D. Hager, L. L. Whitcomb, and R. H. Taylor, “Surgical and interventional robotics: Part II,” vol. 15, no. 3, pp. 94–102, conference Name: IEEE Robotics & Automation Magazine. [Online]. Available:

- <https://ieeexplore.ieee.org/document/4624588/?arnumber=4624588>
- [14] O. M. Omisore, S. Han, J. Xiong, H. Li, Z. Li, and L. Wang, “A review on flexible robotic systems for minimally invasive surgery,” vol. 52, no. 1, pp. 631–644, conference Name: IEEE Transactions on Systems, Man, and Cybernetics: Systems. [Online]. Available: <https://ieeexplore.ieee.org/document/9226520>
- [15] (2022) Intuitive | robotic assisted surgical company | about us. Intuitive Surgical. Accessed: 2024-08-01. [Online]. Available: <https://www.intuitive.com/en-us/about-us/company>
- [16] Intuitive announces FDA clearance of fifth-generation robotic system, da vinci 5 | intuitive surgical. [Online]. Available: <https://isrg.intuitive.com/news-releases/news-release-details/intuitive-announces-fda-clearance-fifth-generation-robotic/>
- [17] D. M. Kwartowitz, S. D. Herrell, and R. L. Galloway, “Toward image-guided robotic surgery: determining intrinsic accuracy of the da vinci robot,” vol. 1, no. 3, pp. 157–165. [Online]. Available: <https://doi.org/10.1007/s11548-006-0047-3>
- [18] J. M. Ferguson, E. B. Pitt, A. Kuntz, J. Granna, N. L. Kavoussi, N. Nimmagadda, E. J. Barth, S. D. Herrell, and R. J. Webster, “Comparing the accuracy of the da vinci xi and da vinci si for image guidance and automation,” vol. 16, no. 6, pp. 1–10. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9540612/>
- [19] SYSTEM | hinotori robotic assisted surgery system | PRODUCT | medicaroid. [Online]. Available: <https://www.medicaroid.com/en/product/hinotori/>
- [20] H. Coussons, J. Feldstein, and S. McCarus, “Senhance surgical system in benign hysterectomy: A real-world comparative assessment of case times and instrument costs versus da vinci robotics and laparoscopic-assisted vaginal hysterectomy procedures,” vol. 17, no. 4, p. e2261. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1002/rcs.2261>
- [21] PRECEYES surgical system -preceyes BV. [Online]. Available: <https://www.preceyes.nl/preceyes-surgical-system/>
- [22] Eye surgery robotics company AcuSurgical successfully completes first clinical study. [Online]. Available: <https://www.medtechdive.com/press-release/>

- 20240704-eye-surgery-robotics-company-acusurgical-successfully-completes-first-clin/
- [23] Microsure MUSA-3. [Online]. Available: <https://microsure.nl/musa/>
- [24] J. Zhao, M. M. Marinho, and K. Harada, “Toward automation of looping in robot assisted pediatric thoracoscopic surgery—study of robotic suture looping with multi-DOF surgical tools—,” vol. 2023, pp. 2P1–B24.
- [25] E. Zhao, M. M. Marinho, and K. Harada, “Autonomous robotic drilling system for mice cranial window creation,” *CoRR*, vol. abs/2406.14135, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2406.14135>
- [26] Y. Koyama, M. M. Marinho, M. Mitsuishi, and K. Harada, “Autonomous coordinated control of the light guide for positioning in vitreoretinal surgery,” *CoRR*, vol. abs/2107.11985, 2021. [Online]. Available: <https://arxiv.org/abs/2107.11985>
- [27] Y. Koyama, M. M. Marinho, and K. Harada, “Vitreoretinal surgical robotic system with autonomous orbital manipulation using vector-field inequalities,” *CoRR*, vol. abs/2302.05567, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2302.05567>
- [28] M. M. Marinho, K. Harada, A. Morita, and M. Mitsuishi, “SmartArm: Integration and validation of a versatile surgical robotic system for constrained workspaces,” vol. 16, no. 2, p. e2053.
- [29] T. Haidegger, “Autonomy for surgical robots: Concepts and paradigms,” vol. 1, no. 2, pp. 65–76, conference Name: IEEE Transactions on Medical Robotics and Bionics. [Online]. Available: <https://ieeexplore.ieee.org/document/8698847>
- [30] Task control with remote center of motion constraint for minimally invasive robotic surgery | IEEE conference publication | IEEE xplore. [Online]. Available: <https://ieeexplore.ieee.org/document/6631412>
- [31] H. Azimian, R. V. Patel, and M. D. Naish, “On constrained manipulation in robotics-assisted minimally invasive surgery,” in *2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pp. 650–655, ISSN: 2155-1782. [Online]. Available: <https://ieeexplore.ieee.org/document/5627985>
- [32] 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.” [Online].

- Available: <http://arxiv.org/abs/2011.04813>
- [33] Automating surgical peg transfer: Calibration with deep learning can exceed speed, accuracy, and consistency of humans | IEEE journals & magazine | IEEE xplore. [Online]. Available: <https://ieeexplore.ieee.org/document/9772013>
- [34] A. Nakazawa, K. Nanri, K. Harada, S. Tanaka, H. Nukariya, Y. Kurose, N. Shono, H. Nakatomi, A. Morita, E. Watanabe, N. Sugita, and M. Mitsuishi, “Feedback methods for collision avoidance using virtual fixtures for robotic neurosurgery in deep and narrow spaces,” in *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, pp. 247–252, ISSN: 2155-1782. [Online]. Available: <https://ieeexplore.ieee.org/document/7523632>
- [35] G. Dagnino, J. Liu, M. E. M. K. Abdelaziz, W. Chi, C. Riga, and G.-Z. Yang, “Haptic feedback and dynamic active constraints for robot-assisted endovascular catheterization,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1770–1775, ISSN: 2153-0866. [Online]. Available: <https://ieeexplore.ieee.org/document/8593628>
- [36] M. M. Marinho, B. V. Adorno, K. Harada, and M. Mitsuishi, “Dynamic active constraints for surgical robots using vector field inequalities,” vol. 35, no. 5, pp. 1166–1185. [Online]. Available: <http://arxiv.org/abs/1804.11270>
- [37] N. Shiraki, T. Wakabayashi, Y. Ikuno, N. Matsumura, S. Sato, H. Sakaguchi, and K. Nishida, “Fovea-sparing versus standard internal limiting membrane peeling for myopic traction maculopathy: A study of 102 consecutive cases,” vol. 4, no. 12, pp. 1170–1180. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2468653020302086>
- [38] D. Zheng, Z. Huang, Q. Zeng, Y. Wang, S. Chen, J. Yi, D. Fang, D. Huang, and W. Chen, “Anatomical and visual outcomes of fovea-sparing internal limiting membrane peeling with or without inverted flap technique for myopic foveoschisis,” vol. 22, no. 1, pp. 1–8, number: 1 Publisher: BioMed Central. [Online]. Available: <https://bmcpophthalmol.biomedcentral.com/articles/10.1186/s12886-022-02679-2>
- [39] M. M. Marinho, B. V. Adorno, K. Harada, K. Deie, A. Deguet, P. Kazanzides, R. H.

- Taylor, and M. Mitsuishi, “A unified framework for the teleoperation of surgical robots in constrained workspaces,” in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 2721–2727, ISSN: 2577-087X. [Online]. Available: <https://ieeexplore.ieee.org/document/8794363>
- [40] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc. [Online]. Available: <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>
- [41] J. Denavit and R. S. Hartenberg, “A kinematic notation for lower-pair mechanisms based on matrices,” *Journal of Applied Mechanics*, vol. 22, no. 2, pp. 215–221. [Online]. Available: <https://doi.org/10.1115/1.4011045>
- [42] B. Vilhena Adorno, “Robot kinematic modeling and control based on dual quaternion algebra —part i: Fundamentals.” Feb. 2017. [Online]. Available: <https://hal.science/hal-01478225>
- [43] Clifford, “Preliminary sketch of biquaternions,” vol. s1-4, no. 1, pp. 381–395, _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1112/plms/s1-4.1.381>. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1112/plms/s1-4.1.381>
- [44] W. R. Hamilton, “II. on quaternions; or on a new system of imaginaries in algebra,” vol. 25, no. 163, pp. 10–13, publisher: Taylor & Francis _eprint: <https://doi.org/10.1080/14786444408644923>. [Online]. Available: <https://doi.org/10.1080/14786444408644923>
- [45] V. M. Gonçalves, P. Fraisse, A. Crosnier, and B. V. Adorno, “Parsimonious kinematic control of highly redundant robots,” vol. 1, no. 1, pp. 65–72, conference Name: IEEE Robotics and Automation Letters. [Online]. Available: <https://ieeexplore.ieee.org/document/7348665>
- [46] H. T. M. Kussaba, L. F. C. Figueiredo, J. Y. Ishihara, and B. V. Adorno, “Hybrid

- kinematic control for rigid body pose stabilization using dual quaternions,” vol. 354, no. 7, pp. 2769–2787. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0016003217300522>
- [47] M. M. Marinho, H. Ishida, K. Harada, K. Deie, and M. Mitsuishi, “Virtual fixture assistance for suturing in robot-aided pediatric endoscopic surgery,” vol. 5, no. 2, pp. 524–531, conference Name: IEEE Robotics and Automation Letters. [Online]. Available: <https://ieeexplore.ieee.org/document/8949747>