Tele-operation Control of a High Dexterity Robot for Vitreoretinal Surgery

Computer Integrated Surgery II   Spring, 2023

Team 21: *Xiao Wei* **Mentors:** *Mojtaba Esfandiari, Adnan Munawar, Dr. Iulian Iordachita*

Introduction

Motivation

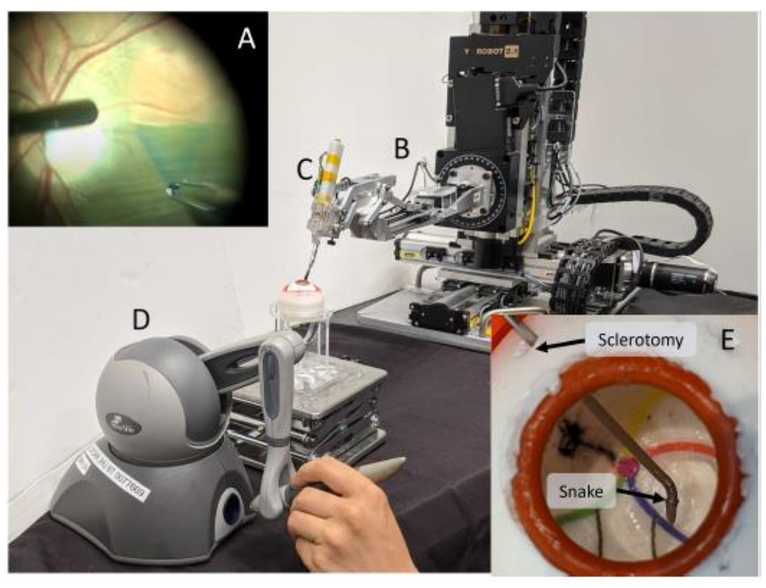
Vitreoretinal surgery requires exceptional precision on micron scale targets in a fragile workspace.

- physiological hand tremor, causing blindness
- forces > 7.5 mN that are imperceptible by human can result in retinal tears

"The peripheral aspects of vitreoretinal surgery," Centre for Sight, 07-Mar-2020. [Online]. https://www.centreforsight.net/blog/the-peripheral-aspects-of-vitreoretinal-surgery/.

Why simulation?

- To enable researchers to quickly develop, validate and test control algorithms
- Won't cause damage on the real robots.



Envisioned high dexterity intraocular manipulator
(A) Epiretinal membrane peeling; (B) SHER; (C) I²RIS; (D) Phantom Omni;

Problems

- Develop a simulation model of the integrated system in AMBF
- Enable robots to interact with an eyeball model using AMBF plugins
- Implement teleoperation control using haptic device

Methods

- **AMBF Simulation**

Blender	Build and assemble models of SHER and I²RIS
AMBF Add-on	Configure joints and rigid bodies and generated ADF files for simulator
AMBF Plugins	Use drilling plugins to interact with anatomy

- **Forward Kinematics**

The SHER comprises five actuated degrees of freedom, including three translational DoFs ('X', 'Y', 'Z') and two rotational DoFs ('Roll', 'Pitch').

The I²RIS is a 2 D.O.F continuum robot, with a diameter of 0.9 mm, measuring at ~3 mm fully extended.

$$F = \begin{bmatrix} R(\vec{q}) & \vec{p}(\vec{q}) \\ \vec{0} & 1 \end{bmatrix} = f(\vec{q})$$

$$F = F_{SHER(q_{1-5})} \times F_{I^2RIS(q_{6-7})}$$

- **Manipulator Jacobian Control**

Input: $[\, v \; w\,]_{haptic}$

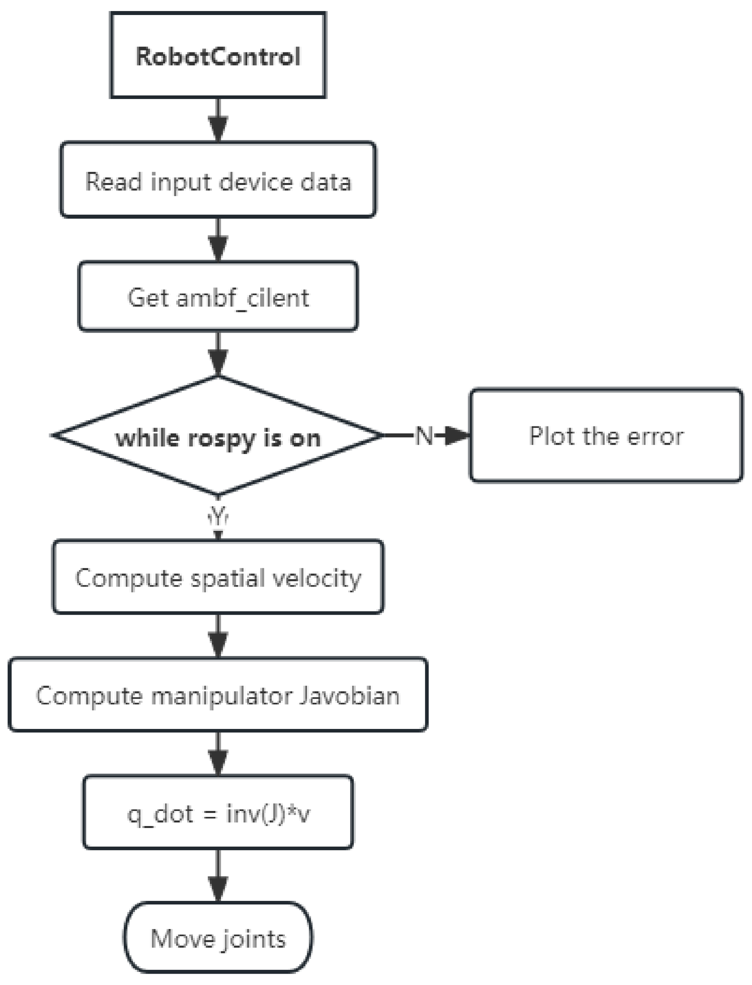
$$F_{haptic\_base} = \begin{bmatrix} R & \vec{p} \\ \vec{0} & 1 \end{bmatrix}$$

$$Ad_g = \begin{bmatrix} R & \hat{p}R \\ 0 & R \end{bmatrix}$$

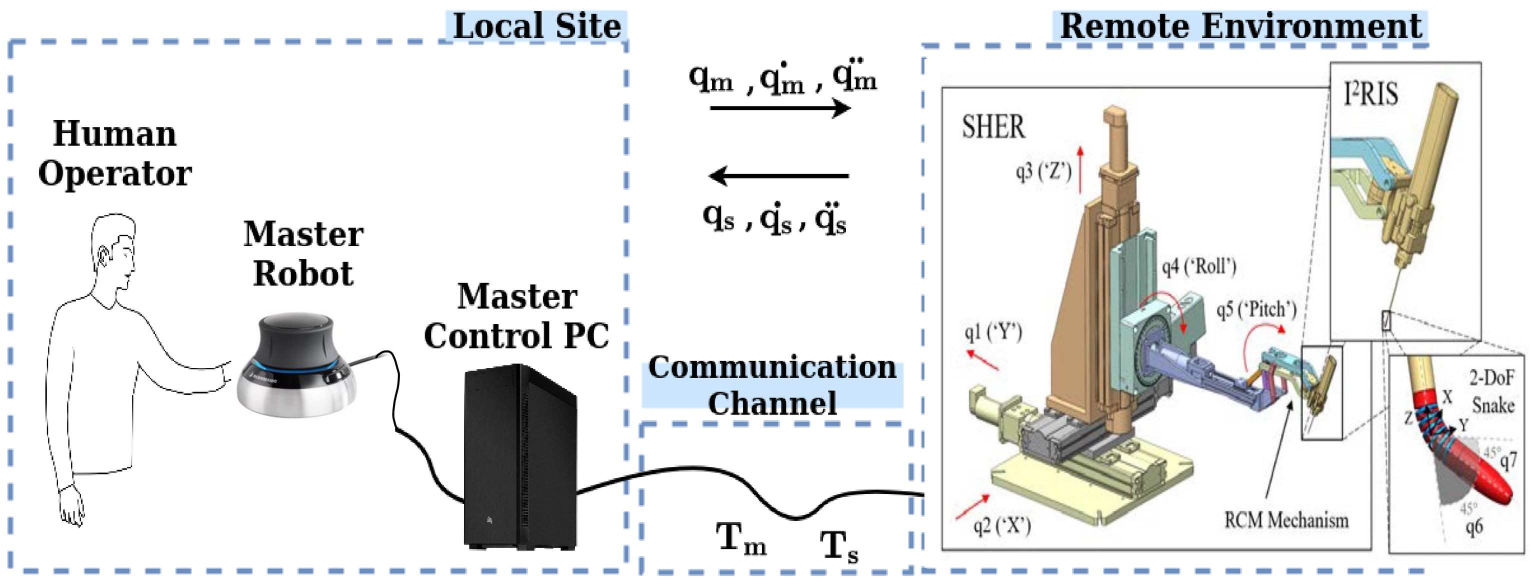
$$[\, v \; w\,]_{spatial} = Ad_g \cdot [\, v \; w\,]_{haptic}$$

$$\dot{q} = J^{-1}(q) \cdot [\, v \; w\,]_{spatial}$$

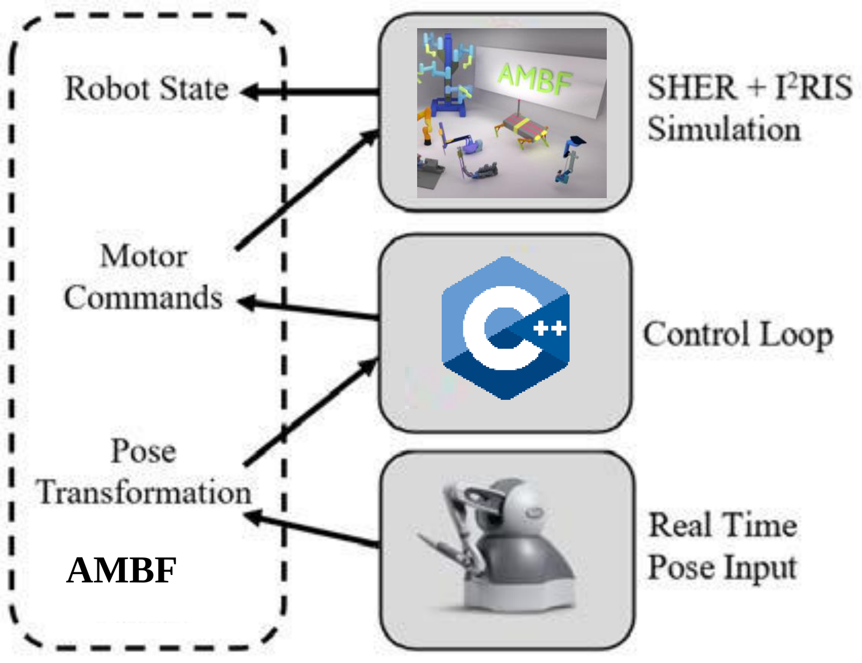
Output: *move joints*



- **Teleoperation Control**



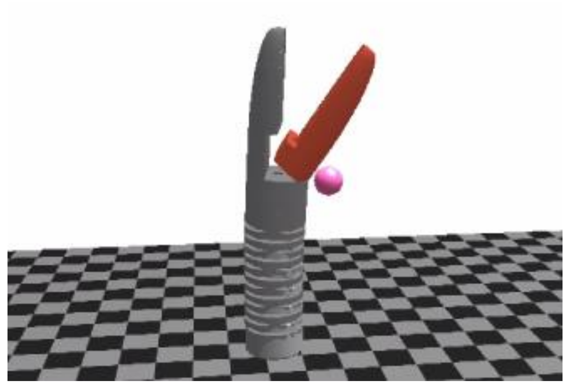
Teleoperation control framework

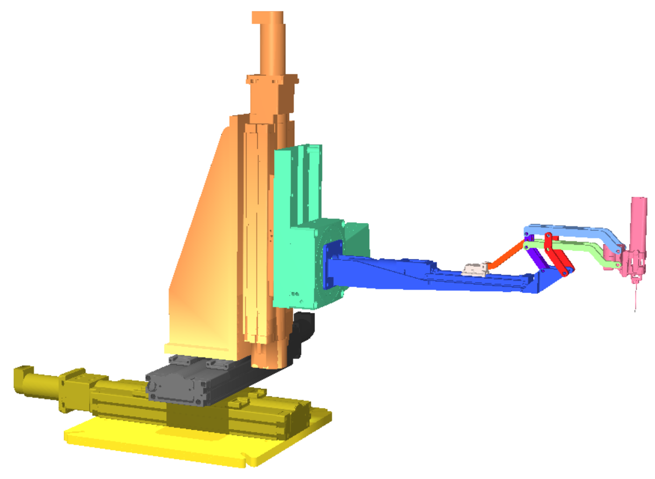


Communication Overview

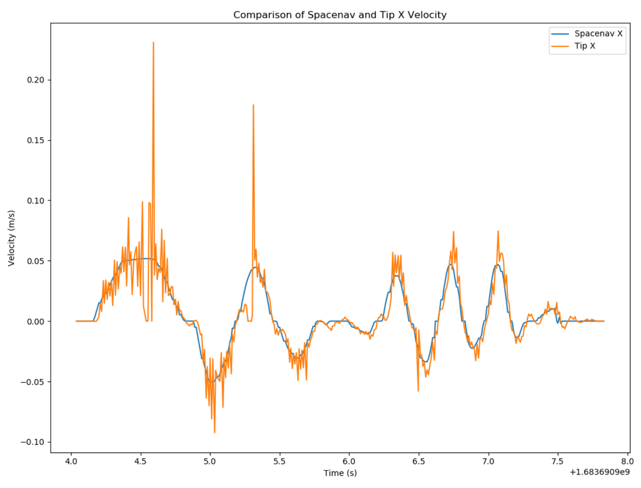
Outcomes and Results

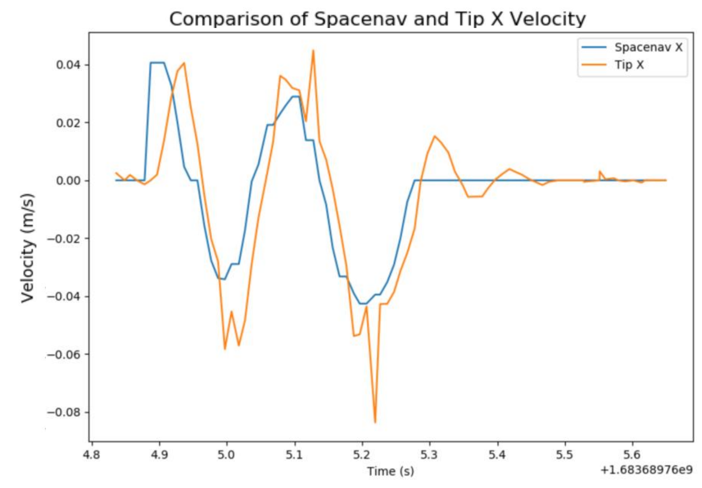
- **Simulation**





- **Real-time Control Error**





- Error is approximately 0.01m/s which is acceptable.

Note: As the working laptop does not support the connection with the Phantom Omni, the 3Dconnexion device will be used as an alternative.

Future Work

- Keep testing algorithms using phantom omni
- Test the controller on real robot systems
- Learn how to design AMBF plugins

Lessons Learned

- Start early for next steps help ensure project success.
- One effective way to learn new things is by trying out examples.
- Effective documentation is crucial for the success of any project.

Credits

- **Xiao Wei**: Responsible for all work.

Acknowledgements

- Thanks to Adnan Munawar, Mojtaba Esfandiari and Dr. Iulian Iordachita

Engineering Research Center for Computer Integrated Surgical Systems and Technology