

A Digital Training Platform for Robotic Vitreoretinal Surgery

Yub Heo^{a*}, Teng Long^{a*}, Mojtaba Esfandiari^a, Haochen Wei^a, Botao Zhao^a,
 Peter Gehlbach^b, Peter Kazanzides^a, Iulian Iordachita^a, and Adnan Munawar^a †

^a*Laboratory for Computational Sensing and Robotics, Johns Hopkins University, Baltimore, MD, 21218, USA,
 (E-mail: yheo1, tlong29, mesfand2, hwei15, bzhao17, pkaz, iordachita, amunawa2@jhu.edu)*

^b*Wilmer Eye Institute, Johns Hopkins School of Medicine, Baltimore, MD, 21287, USA,
 (E-mail: pgelbach@jhmi.edu)*

Vitreoretinal surgery is noted to be extremely challenging due to the precision and scale of manipulation required. Several robotic systems have been developed to overcome these limitations, notably the Steady-Hand Eye Robot (SHER) 3.0, which includes a parallel link base platform and four-bar tilt mechanism to improve performance of the robot in clinical settings. Additionally, teleoperative control has shown improved performance over other control methods in surgical scenarios. However, the delicateness and general inaccessibility of the robot make training on the physical robot to gain familiarity with teleoperation difficult. This work develops a simulation environment for the teleoperation of the SHER 3.0 within the Asynchronous Multi-body Framework (AMBF) to mimic the real robot. A pilot user study is also performed to evaluate the usability and intuitiveness of the simulation environment for a micro-manipulation reach task using teleoperation. Validation results indicate that the simulation environment is functionally similar to the physical robot, which supports its future use as a training platform.

Keywords: Surgical Robotics; Retinal Microsurgery; Digital Twin; Teleoperation.

1. Introduction

Vitreoretinal surgery is among the most sophisticated and technically challenging microsurgical procedures. Conducting procedures within the confined intraocular environment requires exceptionally precise manipulation of microsurgical instruments, where applied forces often fall below the threshold of human tactile perception. Interventions such as epiretinal membrane (ERM) peeling [1] necessitate meticulous control, yet surgeons must overcome restricted depth perception and the difficulty of accurately visualizing the spatial relationship between tools and delicate retinal structures. Furthermore, natural physiological hand tremor substantially challenges the execution of freehand retinal surgery [2].

To mitigate these limitations, a wide range of robotic platforms has been developed, including the Steady-Hand Eye Robot (SHER) [3], hybrid parallel-serial micromanipulators [4], the Preceyes system [5], and the RVRMS platform [6], among others [7, 8, 9]. These systems aim to enhance dexterity, stabilize tool motion by suppress-

ing tremor, and improve surgical precision overall. Control strategies vary across these platforms; some of them employ cooperative manipulation, while others rely exclusively on teleoperation. Notably, teleoperated robotic systems have demonstrated superior performance compared to cooperative counterparts in several surgical scenarios [10, 11]. However, the lack of direct physical coupling between the surgeon and the surgical instrument may introduce challenges related to intuitive control and operator adaptation. In order to improve proficiency in using these surgical robotic platforms via teleoperation, some form of training is required to increase familiarity with their usage. However, given the scarcity of these robotic platforms, and their general delicateness, the direct use of these robots for training purposes is less than ideal.

In this work, we focus on the SHER 3.0 system [12], a five-degree-of-freedom (DoF) robotic manipulator specifically engineered for retinal microsurgery and teleoperated with the Geomagic Touch interface (3D Systems, Rock Hill, SC, USA). The SHER 3.0 consists of several redundant mechanisms, including a parallel-link delta base and a

*These authors contributed equally to this work

†Corresponding Author

2 Author's Name

four-bar linkage tilt mechanism. To facilitate training and promote effective skill acquisition in teleoperated micro-surgical procedures using the SHER 3.0, we developed a high-fidelity virtual training environment using the Asynchronous Multi-Body Framework (AMBF) [13]. Functionally analogous to commercially available simulation platforms such as the da Vinci SimNow (Intuitive Surgical, Sunnyvale, CA, USA), this system integrates the same haptic device utilized by the physical robotic platform, ensuring a representative and seamless transition between virtual training and operative practice.

The primary contributions of this study are summarized as follows:

- (1) A simulation based surgical training simulator that matches the visualization and control mechanics of the physical SHER robot
- (2) Simulation and modeling of a physically and geometrically accurate model of the SHER robot that mimics the redundant kinematics of the physical system
- (3) A pilot user-study testing the usability and control intuitiveness of the simulation system when compared to the physical system on a micro-manipulation reach task using teleoperation

2. Methods

2.1. Robot Kinematics

Fig 1 illustrates the Steady-Hand Eye Robot 3.0 (SHER 3.0), a 5-DoF manipulator including four prismatic joints and a single revolute joint. Due to clinical considerations relating to structural stability and precision [14], spatial restrictions in the operating room, and space for the surgeon [15], several over-constraint mechanisms are employed in the design of the SHER 3.0. Joints J_1 , J_2 , and J_3 are placed such that they control a platform stage via a parallel mechanism. The revolute joint J_3 is used to control the roll of the tool while the prismatic joint J_4 actuate a four-bar linkage to control its pitch. The design does not permit yaw rotation about the instrument's Z axis and its end effector can accommodate a variety of surgical instruments (e.g., a gripper). Figure 1 represents the frames definition of the SHER 3.0, the kinematics of the robot can be described as [16]

$$T_t^b = T_p^b T_r^p T_a^r T_t^a \quad (1)$$

where $\{b\}$, $\{p\}$, $\{r\}$, $\{a\}$ and $\{t\}$ (frames are defined in Fig 1) represent the base, platform, roll, four-bar mechanism, and tip frames respectively. T_p^b is derived from the forward kinematics of the delta mechanism and T_t^a is from the tilting mechanism. Details of kinematic computation and frames are described in [17] and [15]. T_r^p and T_a^r can be described as

$$T_r^p = \text{trans}(Z, l_1) \text{ rot}(X, \theta_1) \quad (2)$$

$$T_a^r = \text{trans}(X, l_2) \text{ trans}(Z, l_3). \quad (3)$$

where *trans* and *rot* denote translation and rotation respectively. l_1 , l_2 , and l_3 denote the translational distances along their respective axes, and θ_1 is the rotational angle about the corresponding axis.

To map the Cartesian velocity of the tool tip into the joint-space velocities of the five actuators, we employ the inverse-Jacobian formulation. The end-effector velocity vector $V_{tb}^b \in \mathbb{R}^6$ (the superscript b indicates that the body velocity in frame $\{t\}$ relative to frame $\{b\}$) can be mapped to the robot joint-velocity vector $\dot{\Theta} \in \mathbb{R}^5$ is expressed as

$$\dot{\Theta} = J_B(\Theta)^{-1} V_{tb}^b \quad (4)$$

where $J_B \in \mathbb{R}^{6 \times 5}$ is the body Jacobian that maps the Cartesian velocity of the tool tip (frame $\{t\}$) to the joint velocities of all five actuators in the base frame $\{b\}$. Joints J_1 – J_3 are three prismatic joints on the delta stage. The derivations of the Jacobian from frame $\{p\}$ (the delta-stage platform) to frame $\{b\}$ (the base) are adopted from [14].

Figure 2 illustrates the kinematic configuration of the 6-DoF haptic device. The Cartesian pose of the handle tip w.r.t. the body-fixed frame $\{B\}$, is streamed in real time to the host computer via the manufacturer's API.

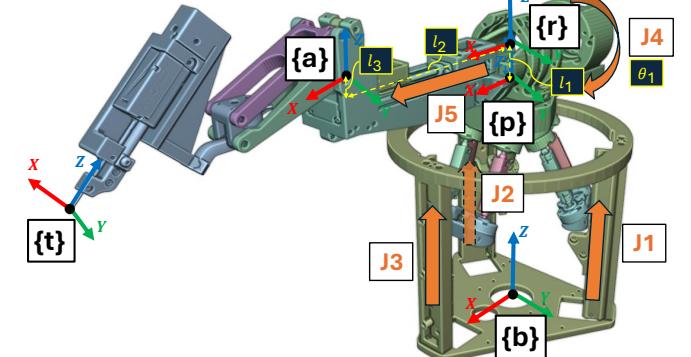


Fig. 1. Kinematic diagram of SHER 3.0

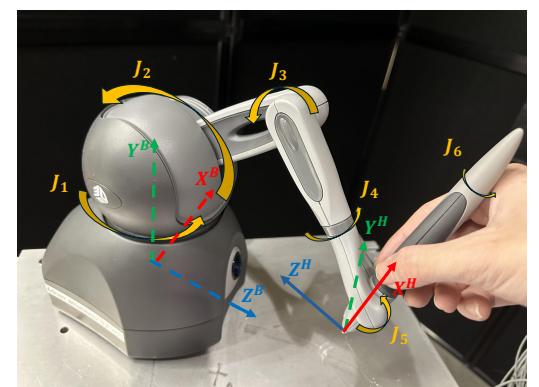


Fig. 2. Kinematic diagram of Geomagic Touch

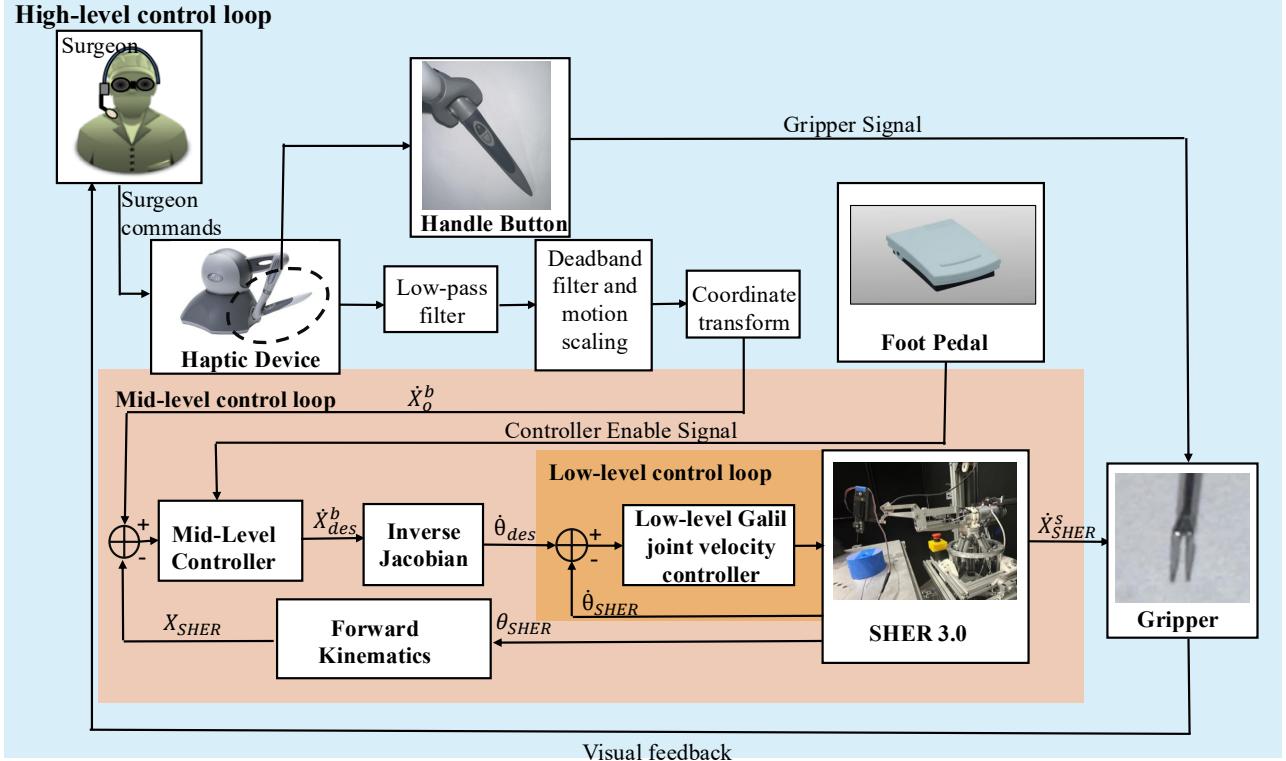


Fig. 3. Block diagram of the teleoperation control architecture. The outer loop implements high-level control with surgeon commands and visual feedback; the middle loop runs the core teleoperation algorithm; the inner loop drives the Galil controller to move the motors.

2.2. Robot Controller

Figure 3 represents the teleoperation control diagram. The user sends velocity commands, \dot{X}_o^b , using the Geomagic Touch haptic device. The end-effector position of SHER, X_{SHER} , is calculated through a forward kinematic function [16] using SHER’s measured joint positions, θ_{SHER} . The error between the haptic device’s inputs and SHER’s end-effector positions is considered as the control error, e . A PID controller, employed as the mid-level control module, is tuned to generate the desired end-effector velocity, \dot{X}_{des}^b , in the SHER’s body-fixed coordinate frame, minimizing the control error. Then, using \dot{X}_{des}^b , the desired joint velocities of SHER, $\dot{\theta}_{des}$, are calculated using an inverse Jacobian function [14]. Then, a low-level joint velocity controller generates this desired joint velocities using an internal PD controller by measuring the actual joint velocities of the robot, $\dot{\theta}_{SHER}$. Finally, the end-effector’s gripper is driven by the Cartesian velocity command \dot{X}_{SHER}^s .

In the high-level control loop, the surgeon observes the gripper’s motion in real time and issues subsequent commands accordingly (see Fig. 3).

In the mid-level control loop, the controller is underpinned by a classical PID algorithm augmented by several auxiliary functions. First, a scaling matrix $M_s \in \mathbb{R}^{5 \times 5}$

extends the effective range of the handle’s motion. Second, the foot pedal provides an enable signal M_p : the controller produces nonzero outputs only when the pedal is engaged. Third, after each clutch activation, the input is re-referenced to the current pose of the handle by a compensation matrix M_c , ensuring the continuity and consistency of the command signal even if the operator moves the handle during pedal disengagement. The controller update law is given by:

$$u[t] = K_P e[t] + K_I \sum_{i=0}^t e[i] \Delta t + K_D \frac{e[t] - e[t-1]}{\Delta t}, \quad (5)$$

where K_P , K_I , and K_D are the PID gains. The error signal $e[t]$ is defined as:

$$e[t] = M_s X_o^s[t] - X_{SHER}[t] - M_c[t], \quad (6)$$

where M_s is the scaling matrix, $X_o^s[t]$ is the pose of the handle tip relative to the haptic-device base frame, $X_{SHER}[t]$ is the gripper pose, and $M_c[t]$ is the compensation offset, defined as:

$$M_c[t] = (M_s X_o^s[t] - X_{SHER}[t]) \mathbf{1}_{\{M_p[t-1]=0, M_p[t]=1\}}, \quad (7)$$

where the $M_p[t] \in \{0, 1\}$ denotes the discrete pedal engagement signal. The compensation offset $M_c[t]$ is nonzero

4 Author's Name

only upon the rising edge of the pedal signal, i.e., when $M_p[t-1] = 0$ and $M_p[t] = 1$.

2.3. Simulation Environment

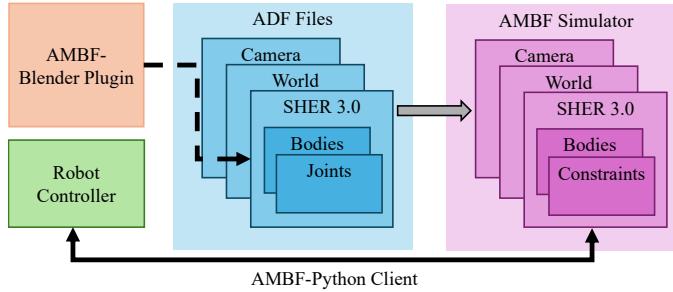


Fig. 4. AMBF Architecture

The modeling of kinematically redundant mechanisms is a key challenge in representing parallel robots in simulation. This is especially true for the SHER 3.0, given the over-constrained design of both the delta-stage platform and tilt mechanism. While several workarounds exist, such as simplification of the parallel robot to a serial one, or the use of features such as *mimic tags* in the popular URDF (Unified Robot Description Format), these workarounds do not realistically model the redundancy and rather visually achieve similar kinematics. Therefore, to avoid such simplifications that convert the redundant robot into an equivalent serial robot, the simulation environment is developed on AMBF [18] to directly implement the parallel robot in simulation.

The use of AMBF allows for real-time dynamic simulation of rigid bodies, joints, cameras, and other scene objects via AMBF Description Format (ADF) files (see Fig. 4). The Blender-AMBF ^a addon is used to create the robot model.

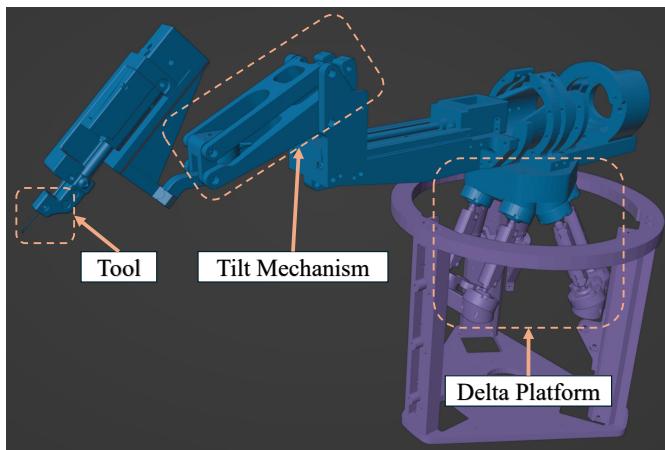


Fig. 5. AMBF Model of SHER 3.0

^ahttps://github.com/WPI-AIM/ambf_addon

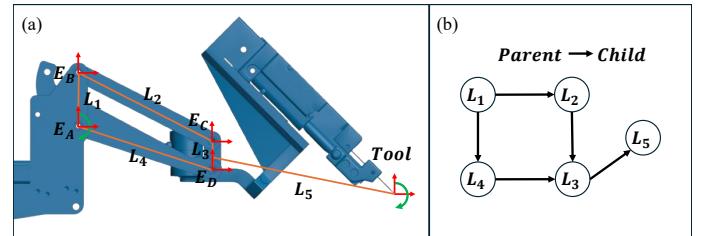


Fig. 6. (a) Four-bar linkage tilt mechanism. L_1, L_2, L_3, L_4, L_5 are linkages of the mechanism. E_A, E_B, E_C, E_D are joints of the mechanism; (b) Interconnected tree structure for tilt mechanism

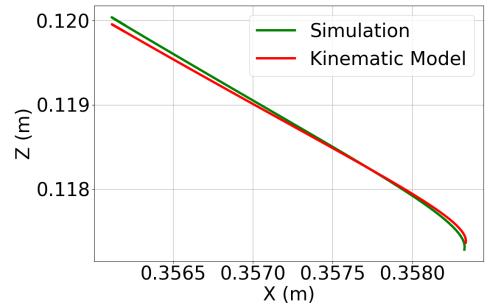


Fig. 7. Error analysis of tilt mechanism

The redundant mechanisms of the SHER 3.0 are modeled as AMBF constraints. Specifically, each link is modeled as a body, and these bodies are connected via a densely interconnected tree structure [19]. This interconnected structure of various bodies allows for each body to have multiple parents or children, which can then be used to model redundant mechanisms mimicking the actual robot. For instance, the tilt mechanism, which is a four-bar linkage, can be modeled as follows (see Fig. 6): each link, L_1, L_2, L_3, L_4, L_5 , are imported as bodies, and each joint, E_A, E_B, E_C, E_D , is modeled as constraint origins between the bodies L_1 to L_4 , L_1 to L_2 , L_2 to L_3 , and L_4 to L_3 , respectively. Additionally, the tool body is set as a fixed constraint with respect to L_3 . When the linkage mechanism is actuated by rotating L_4 about E_A , the constraints imposed allow for the pitch of the tool to be manipulated, similar to the physical robot. Kinematic accuracy of these redundant joints in simulation is subsequently evaluated (see Fig. 7). To evaluate the four-bar linkage, the tilt joint is actuated along its range of motion. The simulation model is deemed to be sufficiently accurate when compared to its kinematics model (mean error = $-2.77 \mu\text{m}$, std. = $2.91e-5$), as it is less than the accuracy requirement of the physical robot ($25-30 \mu\text{m}$) [16].

Commands to control the robot in simulation are communicated through the AMBF Python client, and an optimization-based approach for robot control, as specified in Alg. 2.1, is implemented to control the simulated robot

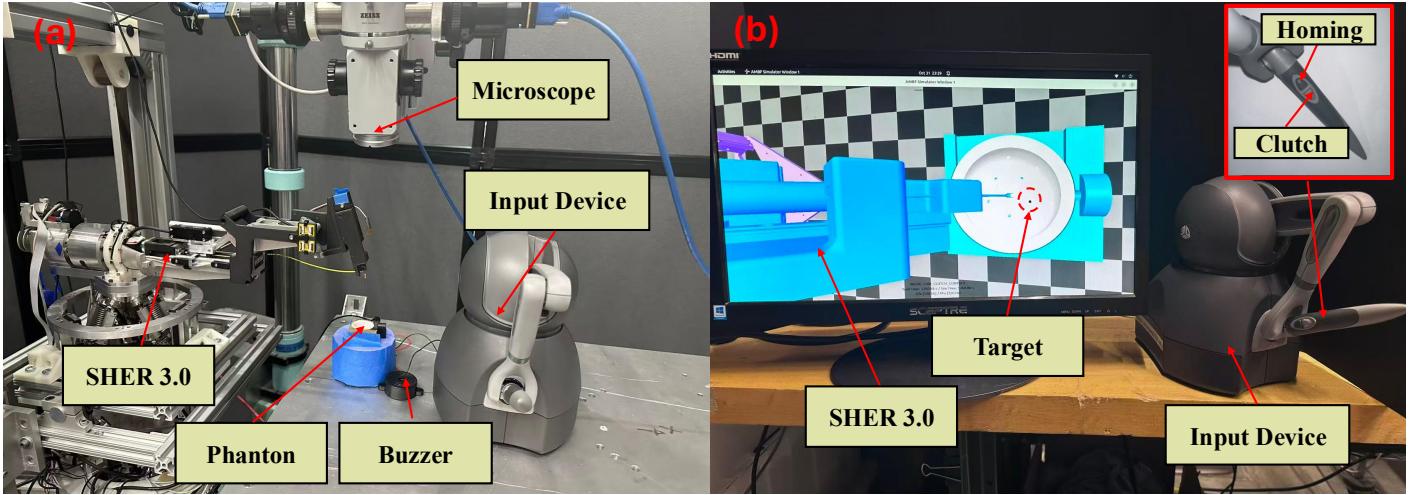


Fig. 8. Experiment Setup (a) Physical robot; (b) Simulation Environment

joints directly and manipulate the tool to its desired position from the haptic device.

Algorithm 2.1. Optimization Based Controller

- 1: Current Tool Pose: X_{curr}
- 2: Desired Tool Pose: X_{des}
- 3: Current Jacobian: $J(q)$
- 4: Joint States: q
- 5: Joint Velocities: \dot{q}
- 6: Objective Function: $C(\dot{q}) = J(q) \cdot \dot{q} - (X_{des} - X_{curr})$
- 7: **while** Running Controller **do**
- 8: Send \dot{q}_{curr} to AMBF client
- 9: Get q_{curr} from AMBF client
- 10: Update X_{curr} via kinematics
- 11: Update $J(q_{curr})$ via kinematics
- 12: Minimize $C(\dot{q}_{curr})$ via Sequential Least Squares Programming method obtain \dot{q}_{curr}
- 13: **end while**

3. Experiments

The experimental aims are to evaluate the fidelity of the proposed simulation environment for the teleoperation of the SHER 3.0 in terms of task completion time, tool trajectory, and control intuitiveness. An insertion task, where the user is required to insert the instrument tip into various holes of a hemispherical phantom, is employed to mimic vitreoretinal surgical tasks.

3.1. Experiment Setup

The experiment setup for the physical robot is presented in Fig. 8(a), and the setup for the simulation environment is presented in Fig. 8(b). The physical setup consists of the real SHER 3.0, custom designed hemispherical phantom, Geomagic Touch, optical microscope (OPMI 1, Carl Zeiss Microscopy, Oberkochen, Germany), and a

buzzer. The hemispherical phantom is of human eye diameter ($D = 24\text{mm}$) and has an inner surface lined with copper foil, which is connected to a buzzer. Insertion holes in the phantom allow for auditory feedback to the user during successful insertion of the instrument tip. The haptic device is used to teleoperate the physical SHER 3.0. The optical microscope is used to get an illuminated, top-down view of the hemispherical surface of the phantom and the robot shaft during manipulation as seen in Fig. 9. Similarly, an AMBF model of the hemispherical phantom is loaded in the simulation environment and placed approximately at a similar location with respect to the robot as in the physical setup. The virtual camera is placed above the phantom to mimic the optical microscope and give the user a top down view of the phantom in simulation as well. The haptic device is used to teleoperate the simulated robot, and the position of the instrument-tip is queried via the AMBF Python Client to determine successful insertion into the holes on the simulated phantom.

3.2. Experiment Procedure

The experiment includes a total of three trials per user for both the simulation environment and the physical robot. Users were randomized and counterbalanced to start with either the simulation environment or the physical robot. Prior to the experiments, each user is given 10 minutes to gain familiarity with the system. Subsequently, the experiment procedure begins with the robot starting at its home position. The user is then instructed to insert the instrument tip into the insertion holes via teleoperation with the haptic handle, following a specified order (see Fig. 9). Fig. 10 shows a typical tool trajectory for the insertion task in both the physical robot and simulation, which is then used to obtain the trajectory length.

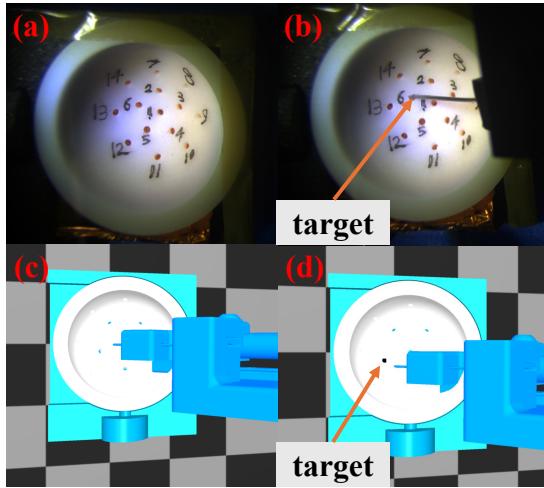


Fig. 9. Top down view during teleoperation task of SHER 3.0. (a) & (b) depict the insertion task in the physical robot while (c) and (d) depict the insertion task in the simulation

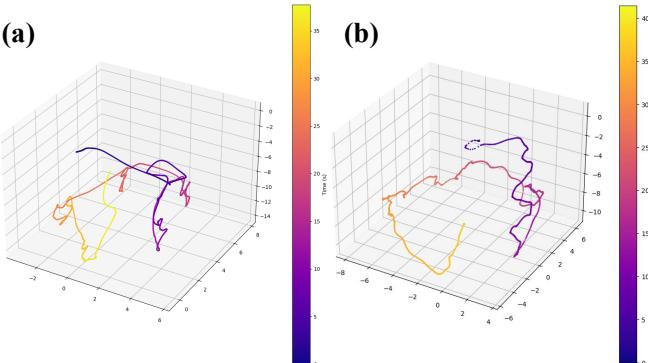


Fig. 10. Trajectory of the tool during teleoperation for the (a) physical robot, (b) simulation

The order is kept consistent between trials, and between the simulator and the physical robot. Afterwards, the user completed a NASA-TLX questionnaire to assess mental workload and user perception. Completion time, number of clutches, and tool trajectory were measured for evaluation as well.

4. Results and Discussion

To evaluate the fidelity of the simulation environment, we compared the results achieved with the physical robot and the simulation environment. Three measures were analyzed: task-completion time, the number of clutch activations (press events: 0→1 transitions), and trajectory length. For each participant, three trials in each environment were conducted, and the group statistics are reported in Table 1. These statistics are used as a form of behavioral benchmarking between the two systems, with similar results indicating that the systems are similar.

Table 1. Comparison of mean completion time, number of clutch activations, and trajectory length for physical robot vs. simulation, for each and all users

Users	Setup	Completion Time (s)	Clutch (count)	Trajectory Length (mm)
User 1	Physical	46.65	3.00	166.78
	Simulation	47.20	3.33	90.82
User 2	Physical	38.11	2.67	114.56
	Simulation	54.17	6.00	132.42
User 3	Physical	40.92	4.67	126.82
	Simulation	41.34	5.67	162.09
User 4	Physical	36.80	2.33	153.02
	Simulation	40.40	2.67	178.16
User 5	Physical	45.63	3.33	103.53
	Simulation	33.47	3.33	100.97
All Users	Physical	41.62	3.20	140.58
	Simulation	43.60	4.57	132.94
		p-value	0.71	0.27
				0.99

Table 1 presents the results of the user study. For all users, a higher *p*-value for all three metrics would indicate that the physical robot and simulation environment are functionally similar, while a lower *p*-value would show inconsistencies between the two environments. Additionally, when comparing mean values, values within a standard deviation of each other would indicate similar task performance. It can be seen that the *p*-value for each of the three evaluation metrics is not statistically significant ($p > 0.7$ for completion time and trajectory length, and $p > 0.2$ for number of clutch activations). It is hypothesized that the lower *p*-value for the number of clutch activations is due to a loss in depth perception within the simulation environment, resulting in a larger number of clutch activations among users. However, it is still not sufficiently low ($p < 0.05$) to be considered as distinct.

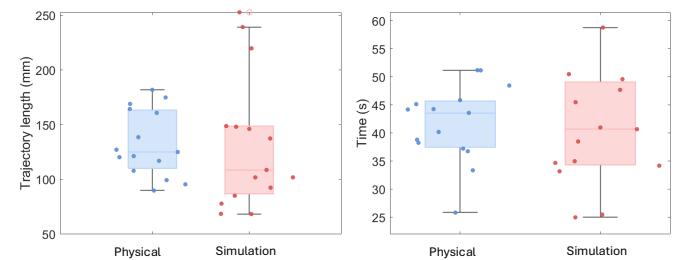


Fig. 11. Comparison of all users' task completion performance between the physical robot and simulation in terms of (a) trajectory length, and (b) completion time

A NASA-TLX questionnaire is also employed to compare the mental load and user perception of the teleoperative insertion task. Fig 12 shows the summarized results of the NASA-TLX questionnaire, which includes six categories: mental demand, physical demand, temporal de-

mand, performance, effort, and frustration. In this questionnaire, their corresponding mean, standard deviation, and p -values are shown in Table 2 to determine whether there is statistical significance. It can be seen that teleoperating the physical robot and the simulation are not statistically significant ($p < 0.05$). Notably, high p -value of 0.90 is observed in the performance criteria. While the mean score for the simulation is slightly elevated for all criteria, this could potentially be linked to reduced depth perception in the simulation, resulting in an increased mental load. However, the results from the NASA-TLX still likely indicate that the simulation has no substantial difference in perceived workload, task difficulty, and user intuitiveness when compared to the physical robot given the p -values.

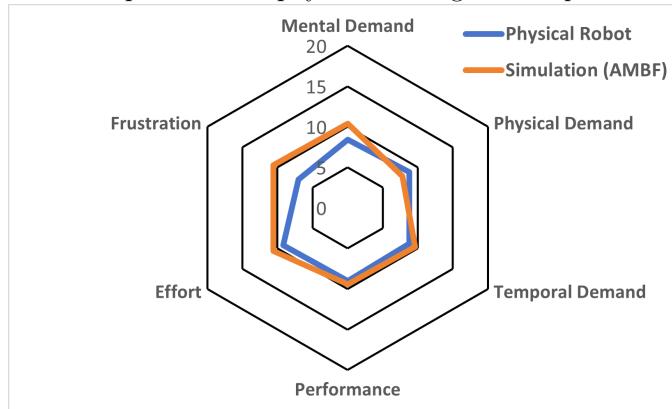


Fig. 12. NASA-TLX questionnaire result

Table 2. NASA-TLX questionnaire (mean \pm std, p -value)

Environment	Physical	Simulation	p -value
Mental Demand	8.4 ± 1.85	10.4 ± 2.24	0.20
Physical Demand	8.8 ± 1.16	7.8 ± 2.40	0.47
Temporal Demand	8.8 ± 2.4	9.6 ± 3.72	0.73
Performance	9.0 ± 5.51	9.4 ± 3.66	0.90
Effort	9.2 ± 3.05	10.6 ± 2.50	0.50
Frustration	7.0 ± 3.58	10.6 ± 3.00	0.16

One limitation of the study is the number of participants (five). This is a pilot study to assess the realism of the simulation environment when compared with the physical robot. As future work, we plan to conduct a multi-user follow-up study involving simulated vitreoretinal surgical tasks, such as epiretinal membrane peeling, to assess the effectiveness of training with the simulation environment.

5. Conclusion

In this work, we developed a high-fidelity simulator of the Steady Hand Eye Robot (SHER) 3.0, a high-dexterity vitreoretinal robotic system with a parallel-link platform and

over-constraint tilt mechanism. The simulator is developed within the Asynchronous Multi-body Framework (AMBF) to allow for the use of maximal coordinates to model the robot constraints. An optimization-based controller is employed to achieve teleoperation of the SHER 3.0 in simulation. The results of the comparison study between the physical robot and simulation indicate kinematic accuracy of the simulator as well as similar levels of task performance and mental load during teleoperation. The findings confirm the intuitiveness and fidelity of the simulation environment when compared to the physical robot and the use of AMBF for control of constrained robotic systems in maximal coordinates. Future work will focus on assessing the effectiveness of training with the simulator for vitreoretinal surgical tasks, such as epiretinal membrane peeling.

Acknowledgments

This work was supported by the U.S. National Institutes of Health under grant numbers R01EB023943 and R01EB034397 and partially by JHU internal funds.

References

- [1] A. T. Fung, J. Galvin and T. Tran, Epiretinal membrane: a review, *Clinical & experimental ophthalmology* **49**(3) (2021) 289–308.
- [2] R. Channa, I. Iordachita and J. T. Handa, Robotic vitreoretinal surgery, *Retina* **37**(7) (2017) 1220–1228.
- [3] A. Üneri, M. A. Balicki, J. Handa, P. Gehlbach, R. H. Taylor and I. Iordachita, New steady-hand eye robot with micro-force sensing for vitreoretinal surgery, *2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, IEEE (2010), pp. 814–819.
- [4] M. A. Nasseri, M. Eder, S. Nair, E. C. Dean, M. Maier, D. Zapp, C. P. Lohmann and A. Knoll, The introduction of a new robot for assistance in ophthalmic surgery, *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, (2013), pp. 5682–5685.
- [5] L. van den Bedem, R. Hendrix, N. Rosielle, M. Steinbuch and H. Nijmeijer, Design of a minimally invasive surgical teleoperated master-slave system with haptic feedback, *2009 International Conference on Mechatronics and Automation*, (2009), pp. 60–65.
- [6] X. Jingjing, H. Long, S. Lijun and Y. Yang, Design and research of a robotic aided system for retinal vascular bypass surgery, *Journal of Medical Devices* **8**(4) (2014) p. 044501.
- [7] A. Gijbels, E. B. Vander Poorten, B. Gorissen, A. Devreker, P. Stalmans and D. Reynaerts, Experimental validation of a robotic comanipulation and telemanipulation system for retinal surgery, *5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*, IEEE (2014), pp. 144–150.

8 REFERENCES

- [8] O. Ergeneman, C. Bergeles, M. P. Kummer, J. J. Abbott and B. J. Nelson, Wireless intraocular microrobots: Opportunities and challenges, *Surgical Robotics: Systems Applications and Visions* (2010) 271–311.
- [9] J. Xiao, Q. Wu, D. Sun, C. He and Y. Chen, Classifications and functions of vitreoretinal surgery assisted robots—a review of the state of the art, *2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*, IEEE (2019), pp. 474–484.
- [10] N. Feizi, M. Tavakoli, R. V. Patel and S. F. Atashzar, Robotics and ai for teleoperation, tele-assessment, and tele-training for surgery in the era of covid-19: Existing challenges, and future vision, *Frontiers in Robotics and AI* **8** (2021) p. 610677.
- [11] T. Osa, S. Uchida, N. Sugita and M. Mitsuishi, Hybrid rate—admittance control with force reflection for safe teleoperated surgery, *IEEE/ASME Transactions on Mechatronics* **20**(5) (2015) 2379–2390.
- [12] A. Alamdar, D. E. Usevitch, J. Wu, R. H. Taylor, P. Gehlbach and I. Iordachita, Steady-hand eye robot 3.0: Optimization and benchtop evaluation for subretinal injection, *IEEE Transactions on Medical Robotics and Bionics* **6**(1) (2024) 135–145.
- [13] A. Munawar, J. Y. Wu, G. S. Fischer, R. H. Taylor and P. Kazanzides, Open simulation environment for learning and practice of robot-assisted surgical suturing, *IEEE Robotics and Automation Letters* **7**(2) (2022) 3843–3850.
- [14] B. Xiao, A. Alamdar, K. Song, A. Ebrahimi, P. Gehlbach, R. H. Taylor and I. Iordachita, Delta robot kinematic calibration for precise robot-assisted retinal surgery, *2022 International Symposium on Medical Robotics (ISMR)*, IEEE (2022), pp. 1–7.
- [15] R. Roth, J. Wu, A. Alamdar, R. H. Taylor, P. Gehlbach and I. Iordachita, Towards a clinically optimized tilt mechanism for bilateral micromanipulation with steady-hand eye robot, *2021 International Symposium on Medical Robotics (ISMR)*, IEEE (2021), pp. 1–7.
- [16] A. Alamdar, D. E. Usevitch, J. Wu, R. H. Taylor, P. Gehlbach and I. Iordachita, Steady-hand eye robot 3.0: Optimization and benchtop evaluation for subretinal injection, *IEEE transactions on medical robotics and bionics* **6**(1) (2023) 135–145.
- [17] J. Wu, G. Li, M. Urias, N. A. Patel, Y.-h. Liu, P. Gehlbach, R. H. Taylor and I. Iordachita, An optimized tilt mechanism for a new steady-hand eye robot, *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE (2020), pp. 3105–3111.
- [18] A. Munawar and G. S. Fischer, An asynchronous multi-body simulation framework for real-time dynamics, haptics and learning with application to surgical robots, *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (2019), pp. 6268–6275.
- [19] A. Munawar, Y. Wang, R. Gondokaryono and G. S. Fischer, A real-time dynamic simulator and an associated front-end representation format for simulating complex robots and environments, *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE (2019), pp. 1875–1882.