2026 IEEE/SICE International Symposium on System Integration (SII)
January 11-14, 2026. Cancún, México

Object-Based Teleoperation Interface for Collaborative Manipulation

Toméo Bourin[1], Tomoya Sasaki[2], Siméon Capy[2], and Eiichi Yoshida[2]

Abstract— **Teleoperation enables human operators to control robots in remote environments, yet its integration with physical human–robot interaction (pHRI) for handling cumbersome objects remains limited. This work introduces a mixed reality (MR) teleoperation interface using an object-based control strategy, enabling the remote operator to manipulate a virtual point on the object rather than the robot's tool center point. The approach was evaluated in collaborative manipulation of an object exceeding the robot's payload capacity and compared with conventional control in a user study. Object-based control supported accurate, differentiated rotational behaviors and was rated more favorably in usability while maintaining low workload, indicating its potential for precise, intuitive manipulation of heavy or bulky objects.**

I. INTRODUCTION

Teleoperation, which enables human operators to control robots in remote environments, is an active research area in industrial, service, caregiving, and assistive robotics. Advances in communication technology have made it possible to achieve real-time interaction over distances exceeding 10,000 km [1]. However, most teleoperation scenarios focus on direct manipulation of the remote robot. Teleoperation that involves collaborative manipulation with a local human, such as physical human–robot interaction (pHRI) [2], remains a relatively unexplored frontier.

One of the most challenging scenarios in human–robot collaborative manipulation is handling cumbersome objects—items that are heavy, bulky, or have awkward geometries—whose mass exceeds the payload capacity of a single robot arm. In pHRI, prior work has investigated approaches such as cobots using force-sensor feedback [3], [4] and dual-arm robots [5], typically without remote control. In these tasks, the robot may guide or support part of the object's motion while the human bears the remaining load. To integrate pHRI and teleoperation, it is essential to design control strategies and interfaces that are both intuitive and well-aligned with the collaborative task.

A common control strategy for robot manipulators uses the tool center point (TCP), which is typically fixed at the robot's end-effector flange and offset to the center of the tool. TCP-based control is well established and widely compatible with various robotic platforms, including industrial robots. However, since TCP-based control was not originally developed for teleoperation, defining control points that are better suited to the task may yield more user-friendly systems. Object-based control is a control strategy that focuses on objects manipulated by robots. It is a promising alternative that offers high operability, as it shares many similarities with the manipulation of virtual objects in 3D design software [6]. Although several methods have been proposed [7]–[9], few studies have examined their application in physical collaborative tasks or evaluated their usability.

[1]National Higher School of Advanced Techniques, 29806 Brest, France
[2]Interactive Robotics Lab, Tokyo University of Science, 125-8585 Katsushika, Japan

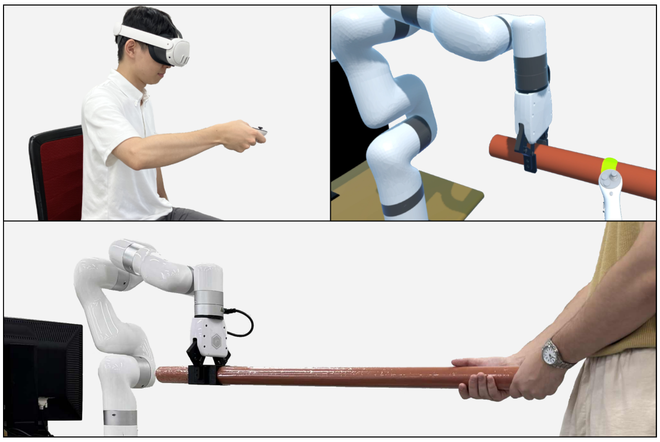


Fig. 1: Object-Based MR Teleoperation and Physical Collaboration with a Local User

For teleoperation interfaces, Mixed Reality (MR) and Virtual Reality (VR) systems have been explored as immersive, spatially coherent tools [10]–[15], providing operators with enhanced perception of the remote environment. While these interfaces have shown benefits in conventional teleoperation tasks, there has been limited investigation into their integration with object-based control for collaborative pHRI. This gap highlights the need for approaches that combine immersive interfaces with control strategies designed specifically for shared manipulation tasks.

In this study, we propose MR teleoperation interface for pHRI, focusing on collaborative manipulation of large or unwieldy objects (see Figure 1). We present the proposed method and its implementation using a robot arm and a head-mounted display (HMD), followed by a technical evaluation and a preliminary user study. The results suggest that our MR-based interface can be operated comparably to conventional methods while offering slightly higher perceived usability.

II. Object-Based Teleoperation Interface

We assume that the robot arms used for teleoperation are joint position control and have a function that calculates inverse kinematics by inputting the target position and orientation. This is because many consumer-available affordable robot arms use position control with servo systems for each joint rather than torque control using force sensors. Such position-based control schemes are also widely applicable to various industrial robot arms. In basic industrial control architectures, motion commands are specified relative to the robot's TCP, often located at the gripper flange. In this section, we first describe the problem setup and a geometric transformation that maps a desired object-level control point to the corresponding TCP position, and then introduce the design of the MR interface required to perform this operation.

A. Problem

Consider a robotic manipulator grasping a rigid object. The TCP (point A) pose in the world frame $\{w\}$ is

$$\boldsymbol{T}_A^w = \begin{bmatrix} \boldsymbol{R}_A & \boldsymbol{p}_A^w \\ \boldsymbol{0}^\top & 1 \end{bmatrix},$$

where $\boldsymbol{R}_A \in SO(3)$ represents the rotation matrix and $\boldsymbol{p}_A^w \in \mathbb{R}^3$ is the position of A. On the grasped object, we define a Virtual Control Point (VCP) M located at a fixed offset from A. The relative pose of M with respect to A is expressed as

$$\boldsymbol{T}_M^A = \begin{bmatrix} \boldsymbol{R}_{AM} & \boldsymbol{p}_{AM}^A \\ \boldsymbol{0}^\top & 1 \end{bmatrix},$$

where $\boldsymbol{R}_{AM} \in SO(3)$ represents the constant local rotation from A to M (fixed due to rigid grasp), and $\boldsymbol{p}_{AM}^A \in \mathbb{R}^3$ is the translation vector from A to M, expressed in frame $\{A\}$.

Given the desired pose of M in the world frame,

$$\boldsymbol{T}_M^w = \begin{bmatrix} \boldsymbol{R}_M & \boldsymbol{p}_M^w \\ \boldsymbol{0}^\top & 1 \end{bmatrix},$$

we aim to relate $\boldsymbol{T}_A^w$ and $\boldsymbol{T}_M^w$ via the known transform $\boldsymbol{T}_M^A$. Figure 2 shows the coordinate frames and transformations.

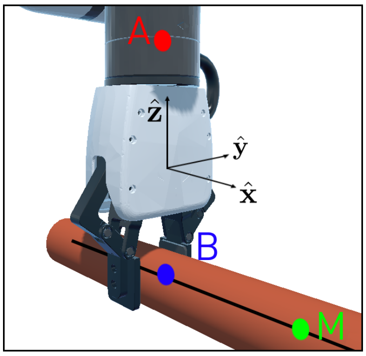


Fig. 2: Reference frames of the robot end-effector and the rigidly grasped object, showing the TCP A, auxiliary point B, and the VCP M.

B. Coordinate Transformation

Here, we first consider the special case where there is no local rotation between the TCP A and the VCP M, i.e., $\boldsymbol{R}_{AM} = \boldsymbol{I}$. In this case, $\boldsymbol{T}_M^A$ represents only a translation.

From rigid-body kinematics, we have the direct relationship between the poses:

$$\boldsymbol{T}_M^w = \boldsymbol{T}_A^w \, \boldsymbol{T}_M^A$$

where the local homogeneous transform $\boldsymbol{T}_M^A$ is defined as:

$$\boldsymbol{T}_M^A = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{p}_{AM}^A \\ 0 & 1 \end{bmatrix}$$

Solving explicitly for the TCP pose, we obtain:

$$\boldsymbol{T}_A^w = \boldsymbol{T}_M^w \, {\boldsymbol{T}_M^A}^{-1}$$

Since $\boldsymbol{T}_M^A$ describes only translation, its inverse is straightforward:

$$(\boldsymbol{T}_M^A)^{-1} = \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{p}_{AM}^A \\ 0 & 1 \end{bmatrix}$$

Expanding the terms, we get:

$$\begin{aligned} \boldsymbol{T}_A^w &= \begin{bmatrix} \boldsymbol{R}_M & \boldsymbol{p}_M^w \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{p}_{AM}^A \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{R}_M & \boldsymbol{p}_M^w - \boldsymbol{R}_M \boldsymbol{p}_{AM}^A \\ 0 & 1 \end{bmatrix} \end{aligned}$$

The positional component can thus be expressed as

$$\boldsymbol{p}_A^w = \boldsymbol{p}_M^w - \boldsymbol{R}_M \, \boldsymbol{p}_{AM}^A.$$

Using this result, the desired TCP pose $(\boldsymbol{p}_A^w, \boldsymbol{R}_A)$ can be directly supplied to the robot's inverse kinematics solver to obtain the corresponding joint commands for teleoperation.

C. Mixed Reality Interface for Control Input

To define the desired pose of the control point M, we design a mixed-reality interface using an HMD and VR controller that provides a life-scale virtual representation of the robot and its surrounding environment, as illustrated in Figure 3, enabling operation from a third-person perspective.

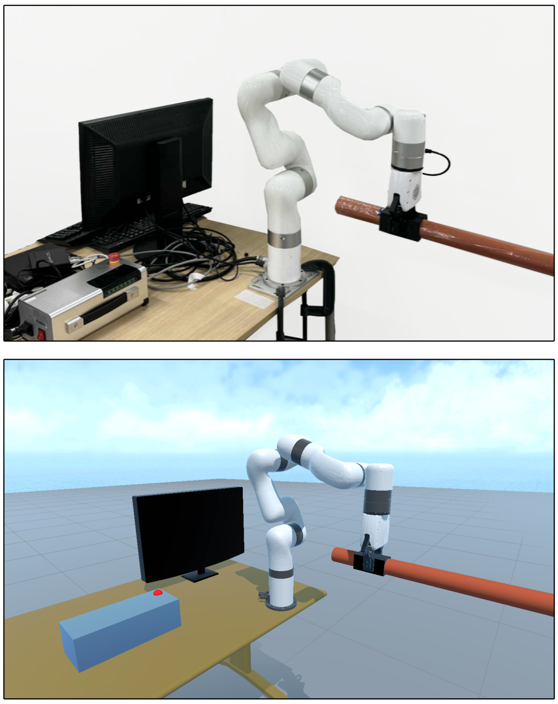


Fig. 3: Comparison between the real environment and the mixed-reality interface.

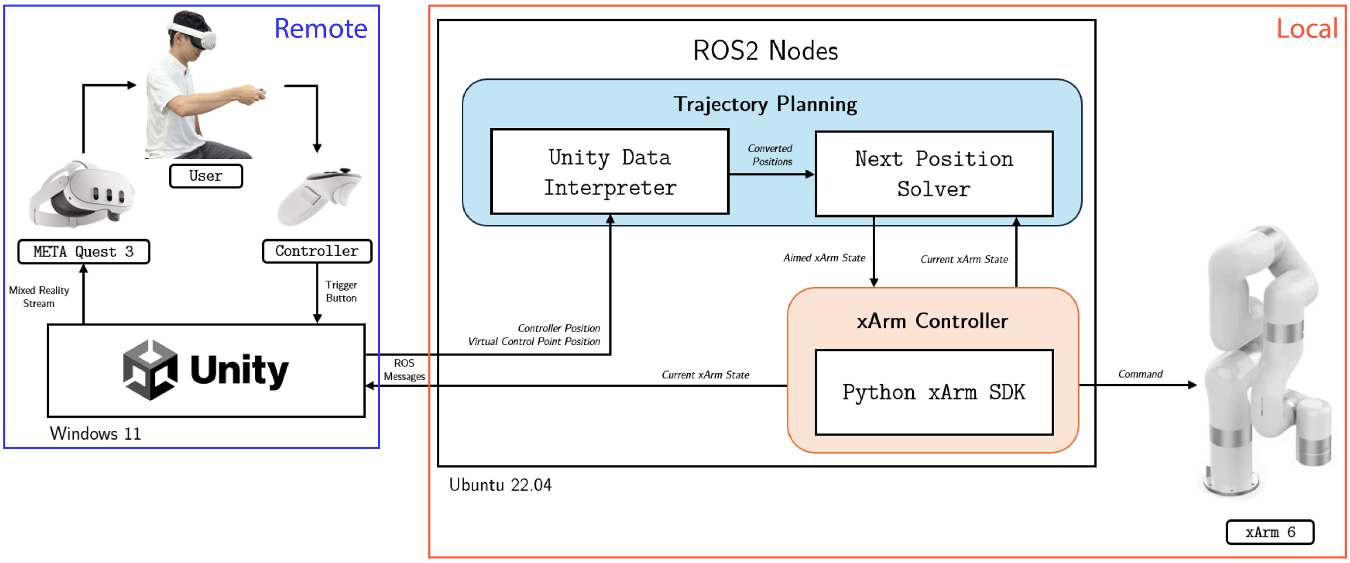


Fig. 4: System architecture overview: (Remote user) a Windows PC runs the MR interface; (Local robot) an Ubuntu PC with ROS 2 controls the robot; both are connected via wireless LAN.

The interface operates as follows: (1) The operator positions the controller near the virtual representation of the object and presses the trigger of the VR controller, prompting the system to register the nearest point on the object as M, thereby defining $\boldsymbol{p}_{AM}^{A}$. If the controller is beyond a predefined threshold distance from the object, the control point defaults to point B (see Figure 2). (2) The operator specifies the object's desired pose by keeping the trigger pressed and performing translational and rotational movements with the controller. The MR system tracks these motions and converts them into the target pose $\boldsymbol{T}_{M}^{w}$.

The computed pose $\boldsymbol{T}_{M}^{w}$ is then transformed into the corresponding TCP pose $\boldsymbol{T}_{A}^{w}$ using the method described in Section II-B. This MR-based interaction approach follows the principles presented by Lee et al. [9].

III. Implementation

We developed a prototype system to verify the proposed method described in Section II. Figure 4 presents an overview of the system architecture, which consists of two main components: (1) a remote user operating through an MR interface, and (2) a local robot arm executing collaborative manipulation tasks. The two components communicate via a wireless local area network (LAN).

A. Remote User

The MR interface is implemented using Unity 6.1 under Windows 11 to generate a life-scale virtual environment. A Meta Quest 3 head-mounted display (HMD) and VR controller are used for visual presentation and user input. All MR computations are executed on a laptop equipped with an Intel Core i7-13700H CPU, 32 GB RAM, and an NVIDIA RTX 4060 GPU.

B. Local Robot

The local manipulator is a UFactory xArm6 (Reach: $700\,\mathrm{mm}$), operated via ROS 2 (Humble distribution) on a desktop PC running Ubuntu 22.04 with an AMD Ryzen 7 PRO 8700GE CPU and 32 GB RAM. Robot motion commands are sent using the native xArm Python SDK. Since the SDK lacks direct velocity control and may cause errors with large displacement commands, we implemented linear interpolation between the current and target TCP poses, as recommended in the User Manual [16]. Interpolated commands are streamed to the robot at $200\,\mathrm{Hz}$ to ensure smooth motion.

C. Network and Synchronization

All components communicate over a wireless LAN. The MR environment is updated at $70\,\mathrm{Hz}$, while ROS 2 handles communication and message passing at $50\,\mathrm{Hz}$. Feedback from the robot, including current joint angles and TCP pose, is continuously transmitted via ROS 2 to the MR interface for real-time synchronization with the digital twin visualization.

IV. Technical Evaluation

The evaluation aims to validate the feasibility of collaboratively manipulating a large and cumbersome object whose weight exceeds the payload capacity of a single robotic manipulator, using the proposed method in cooperation with a human operator. We define tasks involving such overweight objects, manipulated in coordination with a human through object-based control, and record the resulting motions.

A. Experimental Setup

For the experiment, we used a vertical aluminum ladder with a length of $1800\,\mathrm{mm}$ and a total mass of $6.9\,\mathrm{kg}$ as operating object. This weight is heavier than the nominal payload of the xArm6 ($5\,\mathrm{kg}$) [16]. In the task, the xArm6

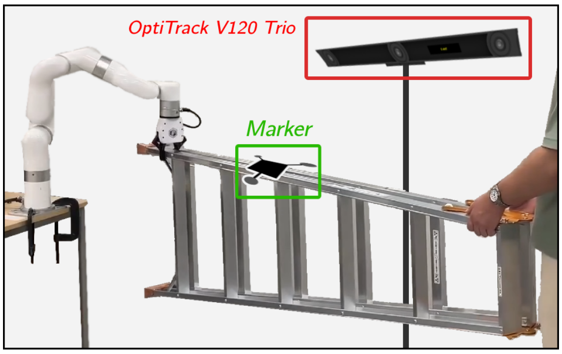


Fig. 5: Configuration of the Motion Capture System

robot grasped one extremity of the ladder, while a human operator grasped the opposite one.

To measure object motion, we used three reflective spherical markers affixed near the ladder's geometric center (see Figure 5). Motion tracking was performed using the OptiTrack V120 Trio[1]. Only the ladder's motion was recorded; the robot and human motions were not directly tracked.

During the evaluation, the robot executed pre-recorded motion trajectories while the human operator was following passively. The experiment was conducted in an environment free of obstacles and workspace constraints.

B. Experimental Protocol

We execute three distinct rotations around the vertical Z-axis to characterize the behavior of the system under different pivot constraints:

- **Motion 1:** Rotation of 35° about the point grasped by the robot.
- **Motion 2:** Rotation of 35° about the location of the motion capture marker.
- **Motion 3:** Rotation of 20° about the point grasped by the human operator. (Note that the rotation angle was limited due to the range constraints of the xArm6.)

This corresponds to performing the same rotational input with different VCPs.

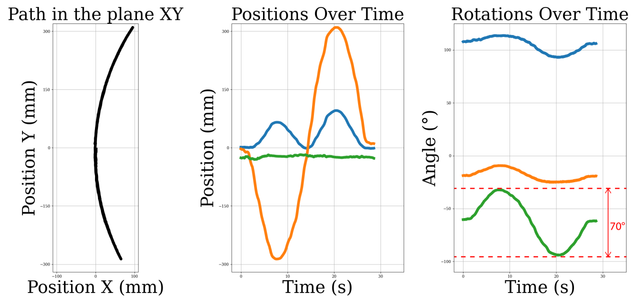
While at rest, we measured the share of the ladder's weight borne by the human. The operator held a scale supporting the object, and we measured approximately 3.70 kg.

C. Results

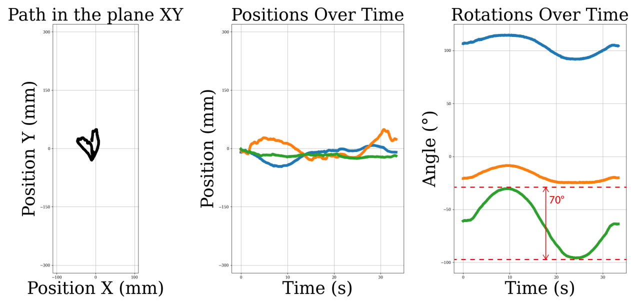
Figure 6 shows the execution results of each motion, from left to right: the X-Y plot, the time series changes in position, and the time series changes in rotation.

In Motion 1 and 3, where the ladder was rotated around one of its extremities, the trajectory in the XY plane follows a smooth arc, and the dominant rotational component occurs around the Z-axis. These results are consistent with a controlled in-plane rotation. In Motion 2, we can observe a minimal displacement with a far-dominant rotation composant around the Z-axis, confirming that the marker closely

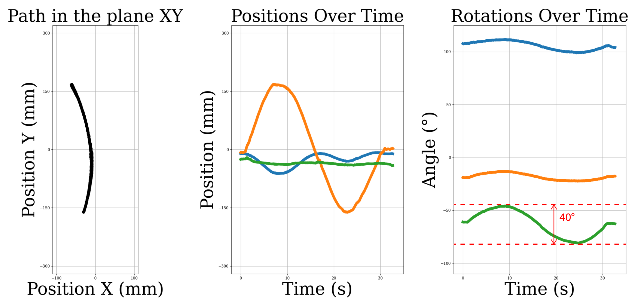
[1]https://www.optitrack.com/cameras/trio-3/indepth.html



(a) Motion 1 data - *Blue: X axis, Orange: Y axis, Green: Z axis.*



(b) Motion 2 data - *Blue: X axis, Orange: Y axis, Green: Z axis.*



(c) Motion 3 data - *Blue: X axis, Orange: Y axis, Green: Z axis.*

Fig. 6: Measurement results for each motion.

approximates the rotational center. However, the path was not perfectly static because the ladder was not rigidly grasped by the robot, and the operator could introduce some bias depending on how well they followed the robot's motion.

Thus, these results confirm that the system can produce differentiated and controlled pivoting behaviors depending on the defined rotational reference.

V. User Study

We conduct a user study to investigate the usability of the proposed method. We compare our Object-Based (OB) control method with a baseline TCP control method (BL), typically used for teleoperating robots. The evaluation consisted of a series of positioning and orientation tasks designed to simulate collaborative work scenarios.

A. Procedure

We employed a within-subject design in which each participant tested both control methods. The order of interface

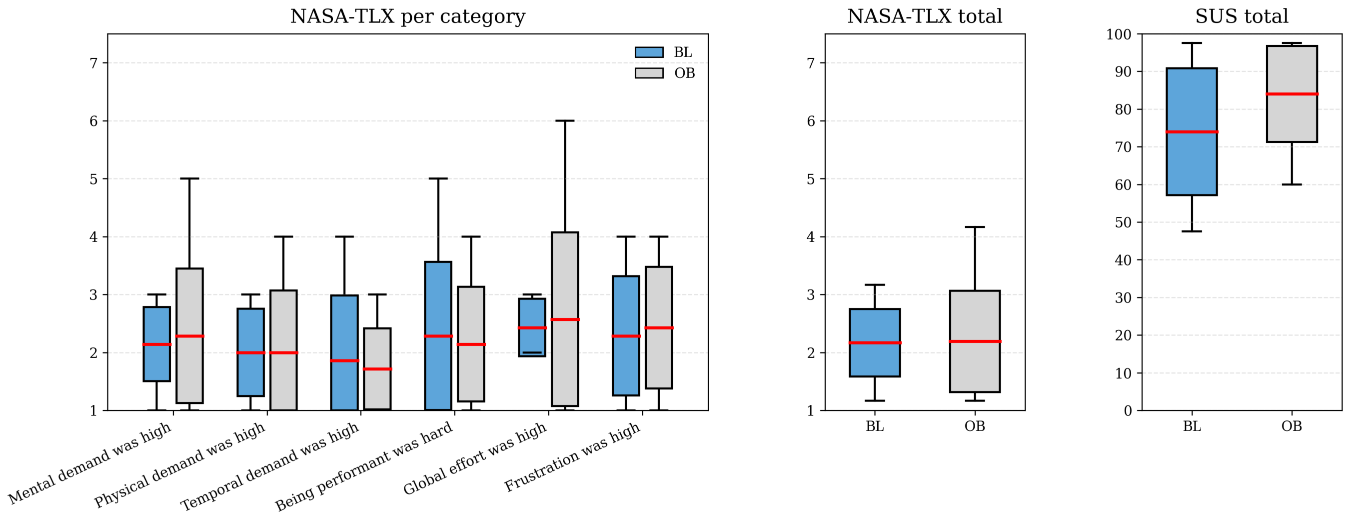


Fig. 7: Comparison of Workload (NASA-TLX) and Usability (SUS) between BL and OB

presentation was shuffled across participants to control for learning and fatigue effects, resulting in two possible sequences (BL→OB or OB→BL). Both orders were equally represented across participants.

For each interface, participants received a brief tutorial followed by a 5-minute practice session. Then they completed 12 predetermined object-placement tasks per interface. After completing a condition, participants filled out the System Usability Scale (SUS) [17] and the NASA Task Load Index (NASA-TLX) [18]. Table I shows the experimental summary.

TABLE I: Experimental Procedure Summary

Phase	Time	Description
Briefing & Setup	5 min	Instructions
Free Practice	5 min	Unrecorded exploration
Block 1	∼5 min	Tasks using Interface 1 (BL or OB)
Questionnaire 1	∼15 min	SUS + NASA-TLX (Interface 1)
Block 2	∼5 min	Same tasks using Interface 2
Questionnaire 2	∼15 min	SUS + NASA-TLX (Interface 2)

B. *Task*

Participants use each interface to align a physical object, manipulated by the xArm6 robot, with a virtual target pose displayed in the MR headset. The object is supported passively by a local human assistant, who helps counterbalance gravity but does not guide the motion as shown in Figure 1. This setup allows the remote operator to focus entirely on the control. Note that these positioning and orientation tasks are performed under the assumption of an obstacle-free environment.

Participants are required to complete each target pose in a fixed sequence to eliminate the influence of a random order.

C. *Measurement*

For both methods, timestamped position and orientation data of the manipulated object are recorded continuously. Additional logs capture pivot selection events and trigger activations to analyze interaction patterns. Quantitative performance is measured via task completion time, defined as the interval between the target pose appearance and successful alignment.

D. *Participants*

7 volunteers from Tokyo University of Science participated in the study. All participants are healthy adult males and novices at robot teleoperation.

E. *Results*

1) Task Performance: The OB method resulted in a longer total completion time (mean = 197.33 s, SD = 25.60 s) compared to the BL method (mean = 171.65 s, SD = 43.99 s), an increase of 25.68 s (+14.96%). With OB, participants often needed to adjust their body posture, sometimes standing up, to reach the object and manipulate the virtual control point.

Beyond these adjustments, OB appeared to encourage a more precise strategy: participants tended to approach the target slowly, refining the object's orientation step by step. This behavior likely contributed to the higher number of Trigger button presses in OB (mean = 50.33, SD = 15.18) relative to BL (mean = 45.00, SD = 4.00), an increase of 5.33 transitions (+11.84%). In contrast, the BL method relied on a single control point and often elicited faster, more trial-and-error–based interactions. Such differences in manipulation strategy may have affected the completion time.

2) Subjective Taskload: The form results indicate that the NASA-TLX —evaluated on a 7-point Likert scale— workload ratings were similar between the object-based and baseline TCP control methods (see Figure 7). Mean scores for each workload dimension remained below the midpoint of the scale for both conditions, suggesting an overall low perceived workload.

In contrast, the System Usability Scale (SUS) scores differed between the two conditions, with OB obtaining a higher average score (84.00 ± 12.71) than BL (74.00 ± 16.85). Note that the commonly accepted usability threshold is 68 out of 100. As participants used the same MR interface in both conditions, these results also indicate that the MR interface was effectively usable independent of the control method.

VI. Discussion and Conclusion

The proposed MR interface for teleoperation and its object-based control method demonstrated efficiency, usability, and a low perceived workload in the context of collaboratively manipulating cumbersome objects. These findings suggest that the approach is a viable alternative for remote collaboration scenarios where rotational precision and reference flexibility are critical. Nonetheless, several directions remain for future work.

First, the current evaluation was limited in both task variety and participant numbers. To better assess generalizability, future studies should examine a broader spectrum of collaborative manipulation tasks, including objects of varying sizes, weights, and geometries, and more challenging tasks, such as guiding a rod through a narrow space, and involve a larger, more diverse participant pool in realistic settings. Moreover, the task should be made more demanding so that successful completion cannot be achieved through brute-force strategies, ensuring that the evaluation more faithfully reflects the intended precision requirements.

Second, our system currently focuses on providing the remote operator with information about the robot's state. In collaborative settings, however, awareness of the local human's state is equally critical for safe and efficient interaction. Future work will aim to extend the system to convey key information about the local user's actions, intentions, and physical constraints, thereby enhancing mutual understanding between the remote and local users.

Finally, the integration of haptic feedback between the remote operator and the local human is a promising avenue. Allowing the remote user to perceive forces acting on the robot could improve coordination, enhance safety, and enable more precise control during collaborative tasks. Furthermore, while this study focused on position-controlled robot manipulators for ease of implementation, we believe that our object-based control approach has the potential to be extended to force-controlled systems, offering valuable insights for the development of future collaborative teleoperation technologies.

Acknowledgment

This work was supported by JST SICORP Grant Number JPMJSC2309.

References

[1] S. Capy *et al.*, "From canada to japan: How 10,000 km affect user perception in robot teleoperation," in *2025 IEEE-RAS 24th International Conference on Humanoid Robots (Humanoids)*, 2025, 25–32.

[2] L. Yue *et al.*, "How to achieve Human–Machine interaction by foot gesture recognition: A review," *IEEE Sens. J.*, vol. 23, no. 15, 16 515–16 528, Aug. 2023.

[3] J. E. Colgate, W. Wannasuphoprasit, and M. A. Peshkin, "Cobots: Robots for collaboration with human operators," in *Proceedings of the ASME 1996 International Mechanical Engineering Congress and Exposition*. Atlanta, Georgia, USA: ASME, 1996, 433–439, paper No: IMECE1996-0367.

[4] M. Peshkin and J. E. Colgate, "Cobots," *Industrial Robot: An International Journal*, vol. 26, no. 5, 335–341, 1999.

[5] S. Tarbouriech *et al.*, "Admittance control for collaborative dual-arm manipulation," in *2019 19th International Conference on Advanced Robotics (ICAR)*. IEEE, Dec. 2019, 198–204.

[6] F. Båberg *et al.*, "Adaptive object centered teleoperation control of a mobile manipulator," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, 455–461.

[7] Y. Matsumoto, S. Katsura, and K. Ohnishi, "Dexterous manipulation in constrained bilateral teleoperation using controlled supporting point," *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, 1113–1121, Apr. 2007.

[8] M. A. R. Garcia, R. A. Rojas, and F. Pirri, "Object-centered teleoperation of mobile manipulators with remote center of motion constraint," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, 1745–1752, Apr. 2019.

[9] J. H. Lee *et al.*, "Robot telekinesis: Application of a unimanual and bimanual object manipulation technique to robot control," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2020, 9866–9872.

[10] S. Duan *et al.*, "Semi-autonomous, virtual reality based robotic telemanipulation for the execution of peg-in-hole assembly tasks," in *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*. IEEE, Nov. 2024, 351–358.

[11] M. Moletta *et al.*, "A virtual reality framework for human-robot collaboration in cloth folding," in *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*. IEEE, Dec. 2023, 1–7.

[12] L. S. Yim *et al.*, "WFH-VR: Teleoperating a robot arm to set a dining table across the globe via virtual reality," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2022, 4927–4934.

[13] F. Kennel-Maushart, R. Poranne, and S. Coros, "Multi-arm payload manipulation via mixed reality," in *2022 International Conference on Robotics and Automation (ICRA)*, May 2022, 11 251–11 257.

[14] C. Wang *et al.*, "Task autocorrection for immersive teleoperation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, May 2021, 3949–3955.

[15] J. I. Lipton, A. J. Fay, and D. Rus, "Baxter's homunculus: Virtual reality spaces for teleoperation in manufacturing," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, 179–186, Jan. 2018.

[16] Shenzhen UFactory Co., Ltd., *xArm User Manual, Translated Version (v2.0.0)*, 2023. [Online]. Available: https://www.ufactory.cc/wp-content/uploads/2023/05/xArm-User-Manual-V2.0.0.pdf

[17] J. R. Lewis and J. Sauro, "Item benchmarks for the system usability scale," *Journal of Usability Studies*, vol. 13, no. 3, 158–167, May 2018.

[18] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (task load index): Results of empirical and theoretical research," in *Advances in Psychology*, P. A. Hancock and N. Meshkati, Eds. North-Holland, Jan. 1988, vol. 52, 139–183.