# A New Mixed-Reality-Based Teleoperation System for Telepresence and Maneuverability Enhancement

Da Sun, Andrey Kiselev, Qianfang Liao, Todor Stoyanov, and Amy Loutfi

***Abstract*—Virtual reality (VR) is regarded as a useful tool for teleoperation systems and provides operators with immersive visual feedback on the robot and the environment. However, without any haptic feedback or physical constructions, VR-based teleoperation systems normally suffer from poor maneuverability, and operational faults may be caused in some fine movements. In this article, we employ mixed reality (MR), which combines real and virtual worlds, to develop a novel teleoperation system. A new system design and control algorithms are proposed. For the system design, an MR interface is developed based on a virtual environment augmented with real-time data from the task space with the goal of enhancing the operator's visual perception. To allow the operator to be freely decoupled from the control loop and offload the operator's burden, a new interaction proxy is proposed to control the robot. For the control algorithms, two control modes are introduced to improve the long-distance movements and fine movements of the MR-based teleoperation system. In addition, a set of fuzzy-logic-based methods are proposed to regulate the orientation, position, velocity, and force of the robot to enhance the system's maneuverability and address potential operational faults. A barrier Lyapunov function and a backstepping method are leveraged to design the control laws and simultaneously guarantee the system's stability under state constraints. Experiments conducted using a six-degree-of-freedom robotic arm prove the feasibility of the system.**

***Index Terms*—Force control, motion regulation, telerobotics, virtual reality.**

## I. Introduction

TELEOPERATION allows human operators to perform unreachable or hazardous tasks at a remote distance. With this property, teleoperation is widely used in outer space exploration, underwater operations, nuclear reactors, and minimally invasive surgery [1]. In recent years, teleoperation has been thoroughly explored with regard to passivity/stability maintenance [2]–[5], transparency enhancement [6]–[8], haptic feedback [9]–[14], and disturbance compensation [15]–[17]. However, in regard to utilizing teleoperation to control industrial robots, the practical workspace and degree of freedom (DoF) of the master haptic device usually cannot cover those of an industrial robot due to the asymmetry between the mechanical structures of master and slave robots. Moreover, in most teleoperation studies, visual feedback is seldom considered, and the human operator is assumed to have full vision of both the robot and the environment. In reality, even with an ideal position and force tracking, a teleoperation system can hardly perform remote tasks without considering the visualization issues of the environment and the robot because operators still have difficulties in judging objects' distances or identifying the right object with which to interact [18]. Some studies employed less complex methods utilizing static cameras or monitors to provide monovision feedback [19], [20]. Nevertheless, the human perception of distance is seriously constrained by the lack of stereoscopic vision and by the parallax effect to the extent that enhancing the telepresence of a teleoperation system is still an open issue.

To reinforce human situational awareness of a real environment and simultaneously break the physical restriction on the master mechanism, virtual reality (VR) has long been advocated as a promising technology for teleoperation interfaces. However, the popularization of VR technology has been possible only recently because of the availability of consumer-grade interface hardware, such as gaming head-mounted display (HMD) devices, and increased computing power. Teleoperation applications usually utilize some sort of mixed reality (MR)-based interfaces because they deal with physical objects in one way or another and because parts of physical and virtual worlds need to be mixed together. Because the exact way in which real and virtual objects are blended may vary, the concept of the reality–virtuality continuum, and a variety of MR settings are introduced in [21], in which the authors propose a method to describe a particular implementation of an MR system based on a three-dimensional (3-D) taxonomy, which includes factors such as reality, immersion, and directness; this allows us to account for a multitude of various aspects simultaneously. An interface based on augmented virtuality (AV) is introduced in [22] to improve performance in pick-and-place tasks. Furthermore, this AV interface also allows the gap in performance between experienced and novice operators to be bridged. This result is in accordance with the previously discovered guidelines by Yanco *et al.* [23], who highlight, among other factors, the importance of providing fused sensor data using a single display, which is achieved by using AV in combination with an HMD technology.

Manuscript received September 20, 2018; revised April 18, 2019, June 13, 2019, and September 19, 2019; accepted November 10, 2019. Date of publication January 11, 2020; date of current version January 14, 2020. This work was supported by the Knowledge Foundation (www.kks.se) under the scope of the AMICI Project: Augmented Interaction for Human-Robot Collaborative Tasks in Industrial Environments. This article was recommended by Associate Editor J. Yang. *(Corresponding author: Qianfang Liao.)*

The authors are with the Center for Applied Autonomous Sensor Systems, Örebro University, 702 81 Örebro, Sweden (e-mail: da.sun@oru.se; andrey.kiselev@oru.se; qianfang.liao@oru.se; todor.stoyanov@oru.se; amy.loutfi@oru.se).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2019.2960676

This result is also supported by the research on applications to mobile robot teleoperation conducted by Nielsen *et al.* [24], demonstrating that 3-D interfaces can improve telepresence. In [25], operators' effectiveness is improved in a search-and-rescue scenario by using an HMD while tracking the operator's head, which provides both stereoscopic vision and a parallax effect. In [26], a 3-D scan of a scene is developed to generate a set of potential grasp candidates for objects to provide guidance for teleoperation.

On the basis of the above references, MR-based teleoperation can provide improved telepresence and have larger potential for improving robot performance over traditional teleoperation in several respects. First, MR-based teleoperation can provide the operator with immersive visual feedback about the remote environment. Second, unlike traditional bilateral teleoperation, which struggles with the asymmetry between master and slave robots, an MR interface imposes no physical restriction on the robots' mechanism, the practical workspace, or the robot's DoF. Thus, an MR interface provides larger freedom to teleoperation control. However, the intrinsic drawbacks of the MR interface, that is, lacking any haptic feedback or physical construction, limit its use for precise control and increases the operational difficulty. Hence, how to enhance the system's maneuverability is a challenge.

This article proposes a novel MR-based teleoperation system, in which a new system design and novel control algorithms are developed to enhance the system's telepresence and maneuverability. The contributions of this article are as follows.

1) For system design:
   a) A new MR interface is designed to teleoperate an industrial robot with multiple DoFs; an immersive HMD-based interface is built with head and hand gesture tracking. The proposed MR interface provides immersive visual feedback to enhance the system's telepresence.
   b) Inside the MR interface, an interaction proxy is designed to teleoperate the robot that allows richer and more natural human–environment interactions. Compared with previous teleoperation systems (see, e.g., [27] and [28]) that have physical restrictions on the master controllers and are seriously affected by human motion, the designed interaction proxy gives the operator larger freedom and convenience to manipulate the robot. Thus, the system's maneuverability can be enhanced.
2) For control algorithms:
   a) Two control modes are proposed to enhance the long-distance movement and fine movement of the MR-based teleoperation system. The combination of these two control modes supports the smooth motion of the robot.
   b) Based on these two control modes, a series of fuzzy logic-based control algorithms are proposed to regulate the robot's position, velocity, and force and update its workspace to handle potential operational faults and enhance the robot's maneuverability.

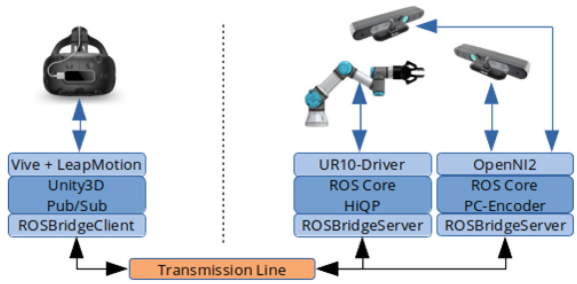


Fig. 1. System architecture. The task space containing the robot and the RGB-D sensor is connected to the MR interface by a transmission line.

   c) A barrier Lyapunov function (BLF) method and a backstepping method are used to design the control laws and to guarantee the stability of the MR-based teleoperation system under state constraints.

## II. MR-Based Interface

The system architecture is presented in Fig. 1. The task space contains a multi-DoF industrial robot and RGB-D sensors. A robot operating system (ROS) is used as a foundation framework for all units in the task space. Furthermore, to allow better system flexibility, the robot and each sensor run their own ROS instances. This allows the introduction of new sensors into the task space without the need to modify other parts and also allows the use of different ROS versions, which is not possible in a master–slave setting.

The MR-based interface is established by virtue of the Unity3D graphics framework. One main merit of the Unity3D graphics framework is the minimization of the development effort in such areas as low-level graphics, display management (in particular, rendering for an HMD), user interaction physics, and animation. However, this framework also introduces challenges in connecting the interface to the robot. The ROSBridge protocol, which provides a socket-based transport for ROS messages between a host system (the robot side) and any third-party software, is applied in this article to connect the MR-based interface to the robot. Therefore, every unit in the task space implements the ROSBridge server, to which the MR interface connects. The server-side ROSBridge package is further developed to implement binary data transfer and message compression to reduce the bandwidth. The interface side implements a ROSBridge connector that publishes a number of messages related to the desired pose, the operator's pose in the environment, the gestures being made, and the status of user interface elements.

The implemented interface applies an AV setting, where the scene contains a model of the robot in a virtual environment, augmented with real-time robot joint states and a point cloud of the task space from an RGB-D sensor. The AV setting is selected to isolate operators from their local environment and direct their attention to the remote task space. Additionally, the interaction paradigm, described in further detail in Section III, is built to allow a third-person view of the scene, where the operator is decoupled from the control loop. To reduce the

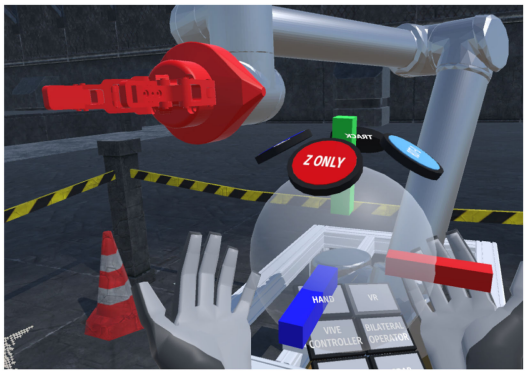


Fig. 2. Interaction proxy in the virtual environment: A transparent sphere with a coordinate frame and some buttons. The coordinate frame is used for targeting the desired pose, and the virtual buttons, which can be freely attached to the interaction proxy, are linked to different functions.

bandwidth, the interface part is subscribed to combined row imagery data from the sensor, transmitted at a rate of 10 frames/s. The point cloud is then reconstructed on the interface part. This allows the bandwidth to be greatly reduced, and selective point cloud rendering can be implemented on the interface side. The bandwidth between the robot and the interface side is 1.5 MB/s, which includes all ROS messages and RGB-D sensor images.

## III. Fuzzy-Logic-Based MR Teleoperation

The main goal of developing the interface in this article is to allow an operator to interact with the environment in a natural way. Accordingly, an HMD is applied to allow the operator to move freely in the virtual task space and to provide stereoscopic vision and a parallax effect. This is achieved by using an HTC Vive HMD in a big-room setting (approximately $3 \times 4$ m). Furthermore, the operator's hands are tracked using a Leap Motion gesture tracker mounted on the HMD. The benefit of this setup is that the operator can interact with the environment using hand gestures. Unlike previous works, the gestures captured by the Leap Motion sensor in this article are not directly used to control the robot since this approach can easily generate tracking errors. Instead, an "interaction proxy" is designed in the MR-based interface to control the position and orientation of the robot to ease the operator's task. The interaction proxy is a virtual object that can be freely rotated and moved by the operator inside the virtual environment, as shown in Fig. 2. Compared with the VR controllers in [22] and [27], where the operator has to physically grasp a handle and cannot let go of it when controlling the robot, a salient advantage of the proposed virtual proxy is that the operator is not required to physically hold anything and can be freely decoupled from the control loop at any moment to take a rest or change a pose to proceed with the next motion. Therefore, the burden of performing tasks on the operator can be alleviated, and the possibility that the robot will be disturbed by a gesture unintentionally made by the operator can be reduced.

However, since the virtual proxy cannot be physically touched, it is easy for the operator to generate operational faults. To resolve this problem and enhance the system's maneuverability, we propose two control modes [coarse movement mode (CMM) and fine movement mode (FMM)] together with a series of robot state regulation strategies. A virtual button is created at the interaction proxy that can switch between "activated" and "deactivated" by pressing the button in the virtual environment, and we use this button to switch between these two control modes. The CMM is designed for the robot's long-distance movement. In this mode, at the beginning, the button is deactivated to lock the robot's motion and make it remain in its current pose so that the operator can freely adjust the position and orientation of the interaction proxy. After placing the interaction proxy in the desired location, the operator can activate the button by pressing it to allow the robot to track the interaction proxy. With this function, the robot does not need to continuously follow the virtual proxy in the CMM, and the operator can have enough time to place the virtual proxy in the desired pose when the button is deactivated. As a result, operational faults can be avoided. On the other hand, the FMM is designed for fine movements by providing the robot with continuous references, where the button is always activated such that the interaction proxy continuously influences the movement of the robot. To prevent operational faults in the FMM, effective strategies are needed to regulate the robot states, including its orientation, position, and velocity. In the remainder of this section, the regulations imposed on the orientation, position, velocity, and force in these two control modes are introduced.

### A. Orientation Regulation

As stated before, in the CMM, by deactivating the button to lock the robot's motion, the operator can freely adjust the interaction proxy without influencing the robot. Therefore, the operator can have sufficient time to determine an appropriate orientation for the robot through the virtual proxy. In the FMM, where the robot continuously tracks the interaction proxy, having the robot perform a task will be difficult to accomplish and will place a heavy burden on the operator since he/she needs to simultaneously take care of both the position and the orientation of a multi-DoF robot. To resolve this issue, at the moment that the button is activated (at the moment FMM is launched), the current reference orientation is recorded as ${}^{o}\mathrm{X}_{\mathrm{str}}$ and is taken as the reference orientation for the FMM. Therefore, in the FMM, the reference orientation is always constant as ${}^{o}\mathrm{X}_{\mathrm{str}}$, and the operator needs only to regulate the position of the robot. ${}^{o}\mathrm{X}_{\mathrm{str}}$ can be updated during the CMM.

### B. Position Regulation

We first introduce the position regulation in the FMM. A fuzzy-logic-based method is applied to regulate the reference position signal, as shown in Fig. 3, where $X_{r1}$ represents the position of the center of the interaction proxy and $X_{r3}$ denotes the current position of the end-effector $X_s$. Two concentric spheres are designed with the center of the spheres located at the tip of the robot. The radii of these spheres are $r$ and $R$ ($R > r$). The two spheres represent the workspaces of the robot in its current states. Then, a straight line is created between $X_{r1}$ and

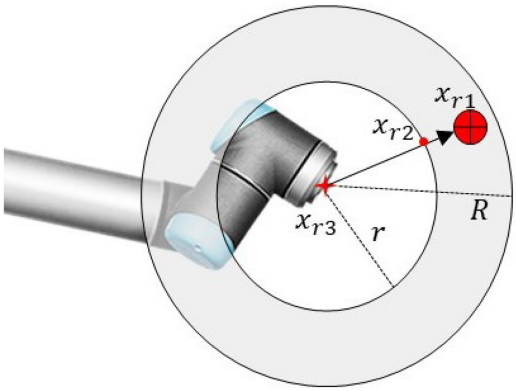


Fig. 3. Definition of the workspaces represented by concentric circles.

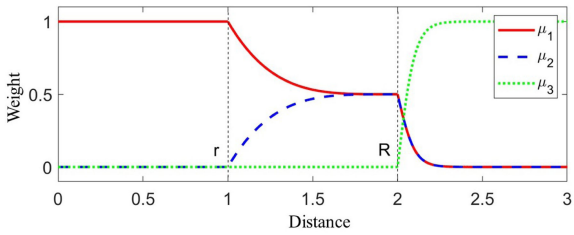


Fig. 4. Example diagram of $\mu_1, \mu_2$, and $\mu_3$ ($r = 1, R = 2, \sigma = 4, \iota = 20$).

$X_{r3}$. $X_{r2}$ is the point of intersection between the line and the small sphere (there are two points of intersection of the straight line with the small sphere; the point closest to $X_{r1}$ is the one we need). The position error between the interaction proxy and the robot is defined as

$$e_p(t) = X_{r1}(t-T) - X_o(t) \tag{1}$$

where $T$ denotes the time delay and $X_o$ denotes the recorded initial position of the robot. Based on $e_p(t)$, we define the fuzzy memberships $\mu_1$, $\mu_2$, and $\mu_3$ to be

$$\mu_1 = \begin{cases} 1, & \text{if } \|e_p\|_2 \le r \\ \frac{1}{2}\left(\frac{R}{R-r} - \frac{1}{R-r}\|e_p\|_2\right)^{\sigma} + \frac{1}{2}, & \text{if } r < \|e_p\|_2 \le R \\ -\frac{1}{2}\tanh\left(\iota\left(\|e_p\|_2 - R\right)\right) + \frac{1}{2}, & \text{if } \|e_p\|_2 > R \end{cases} \tag{2}$$

$$\mu_2 = \begin{cases} 0, & \text{if } \|e_p\|_2 \le r \\ \frac{1}{2} - \frac{1}{2}\left(\frac{R}{R-r} - \frac{1}{R-r}\|e_p\|_2\right)^{\sigma}, & \text{if } r < \|e_p\|_2 \le R \\ -\frac{1}{2}\tanh\left(\iota\left(\|e_p\|_2 - R\right)\right) + \frac{1}{2}, & \text{if } \|e_p\|_2 > R \end{cases} \tag{3}$$

$$\mu_3 = \begin{cases} 0, & \text{if } \|e_p\|_2 \le R \\ \tanh\left(\iota\left(\|e_p\|_2 - R\right)\right), & \text{if } \|e_p\|_2 > R \end{cases} \tag{4}$$

where $\sigma > 1$ and $\iota \gg 1$ are positive constants whose values determine the change rates. $\mu_1, \mu_2$, and $\mu_3$ satisfy $\mu_1 + \mu_2 + \mu_3 = 1$. Fig. 4 shows an example of the above fuzzy memberships. The final reference position can be derived as

$$X_r(t) = \mu_1 X_{r1}(t-T) + \mu_2 X_{r2}(t) + \mu_3 X_{r3}(t). \tag{5}$$

According to (2)–(5) and Fig. 4, when the interaction proxy is inside the small sphere ($||e_p||_2 < r$), the proxy takes full control to drive the robot. When the interaction proxy is between the small sphere and the large sphere ($r \le ||e_p||_2 \le R$), the target pose of the robot is codetermined by $X_{r2}$ and $X_{r1}$. By setting $\sigma > 1$, the control authority of $X_{r1}$ can be quickly transferred to $X_{r1}$ and $X_{r2}$; thus, the robot can be "dragged" by $X_{r2}$ and will not exceed the large workspace. When the interaction proxy is outside the large sphere ($||e_p||_2 > R$), this is defined as an operational fault. By setting $\iota \gg 1$, the robot is basically not controlled by the interaction proxy and will stay at its current position, with $\mu_3$ sharply increasing to 1.

The workspaces will then update when the robot $X_s$ reaches the reference position $X_r$ (the tracking error $e_s$ is small enough). That is, $(||e_s||_2 = ||X_s - X_r(t-T)||_2)^2 \le \xi_s^2$, where $\xi_s$ is a positive constant close to zero. The initial position $X_o$ also updates to be the current robot position, which updates when $||X_s - X_r||_2 \le \xi_s$.

By leveraging the above fuzzy memberships and the update rules, updating the workspace allows the interaction proxy to drive the robot to make a small movement at every time step, which is helpful in the case that a sudden jump by the interaction proxy caused by an operational fault leads the robot to generate an unwanted pose or even causes a singularity.

For the position regulation of the CMM, by setting $r$ to be longer than the radius of the robot's physical workspace, the interaction proxy can have full priority to control the robot to make a long-distance movement.

### C. Velocity Regulation

We set a fuzzy-based velocity boundary for the FMM to be

$$B_{v2} = (b_{u2} - b_{l2})\mu_{v2}^{\sigma_2} + b_{l2} \tag{6}$$

where $b_{u2}$ and $b_{l2}$ are two positive constants representing a large upper bound and a small lower bound, respectively. $\sigma_2 > 1$ represents the change rate. $\mu_{v2}$ is a fuzzy membership defined as

$$\mu_{v2} = \mu_{1,t_{k-1}} \cdot \mu_{1,t_k}. \tag{7}$$

The fuzzy membership $\mu_{v2}$ is the product of $\mu_1$ at the current time step and that at the preceding time step. From (7), $\mu_v = 1$ means that the robot is conducting a small movement in a certain period ($t_k - t_{k-1}$), so its velocity does not degrade. Once the interaction proxy is moved outside the small workspace during this period, $B_{v2}$ is reduced to decrease the velocity. The slower motion of the robot provides the operator with more time to regulate the pose of the interaction proxy.

Note that the MR reference position in the CMM is more like a step signal that can introduce a position discontinuity. Since the interaction proxy can be freely placed in the physical workspace of the robot, the step reference position can be very large. Therefore, the velocity boundary $B_{v1}$ in the CMM should be small at the beginning of the movement and then gradually regulate itself during the movement. Accordingly, the velocity boundary $B_{v1}$ is defined as

$$B_{v1} = \frac{-(b_{u1} - b_{l1})|\mu_{v1} - \frac{1}{2}|^{\sigma_1}}{|\frac{1}{2}|^{\sigma_1}} + b_{u1} \tag{8}$$

where $b_{u1}$ and $b_{l1}$ are the upper bound and lower bound, respectively, and $\mu_{v1}$ and $\sigma_1$ are defined as

$$\mu_{v1} = \frac{||X_s - X_o||_2}{||e_p||_2} \tag{9}$$

$$\sigma_1 = (\sigma_u - \sigma_l) * \mu_{v1} + \sigma_l \tag{10}$$

where $\sigma_u > 1$ and $\sigma_l < 1$ are positive constants.

The logic of (8)–(10) is that when the robot is at its initial position, the fuzzy membership $\mu_{v1}$ is close to zero, which makes the boundary $B_{v1}$ its lower bound. Additionally, the change rate $\sigma_1$ is low at the beginning of motion to prevent the robot from making sudden movements. During a long-distance movement, the velocity boundary increases to its upper bound to accelerate the robot. The change rate $\sigma_1$ also becomes high enough to allow the robot to have a fast movement speed. When the robot is close to its desired pose, $\mu_{v1}$ reaches 1 to slow down its motion by lowering the velocity boundary. The trajectory of $B_{v1}$ is a bell-shaped curve. This allows the robot to stably track the interaction proxy.

### D. Force Regulation

In practical industrial applications, normally, an industrial robot is not used to conduct hard contact motion, since its large applied force is likely to destroy the environment or itself. However, during MR-based teleoperation, in both the CMM and the FMM, it is possible that the robot will strike some rigid environmental object due to an operational fault. Although the proposed position and velocity regulation strategies can prevent the robot from making sudden movements, any hard contact between the robot and an object in the environment may still destabilize the overall system or cause tasks to fail.

To prevent the robot from repetitively hitting an object in the environment due to an incorrect pose taken from the interaction proxy, we design an effective strategy to regulate the robot's force. The estimated external force $F_e^*$ is set to be limited by a constant boundary $B_f$, where any force exceeding $B_f$ is deemed a hard contact that may make the system crash. If the robot strikes a physical barrier when moving to a location set by the interaction proxy, the estimated force will jump close to $B_f$, and the position of the robot at this moment is recorded as $X_{sf}$. Then, the feedback force pushes the robot away from the physical barrier like a spring. With the recorded position $X_{sf}$ and the current robot's position $X_s$, as in the example illustrated in Fig. 5, the center $X_c$ of the concentric circles with radii $r$ and $R$ can be automatically updated (the created concentric circles in Fig. 5 after contacting a physical barrier have the same radii of $r$ and $R$ as the concentric circles in FMM). That is, by using $X_{sf}$ and $X_s$ to generate a straight line, $X_c$ is located along this line and satisfies $||X_{sf} - X_c||_2 = R$ and $||X_s - X_c||_2 \leq ||X_{sf} - X_c||_2$.

In Fig. 5, based on (2)–(5), the interaction proxy $X_{r1}$ is outside the large circle, which means it cannot prevent the robot from striking the physical barrier again. The robot's current position $X_s$ ($X_{r3}$) takes the control priority so that the robot will stay at its current position until the operator correctly regulates the interaction proxy's position. This method can prevent the robot from getting stuck on a hard contact or from striking a physical barrier multiple times due to the spring-like force feedback.

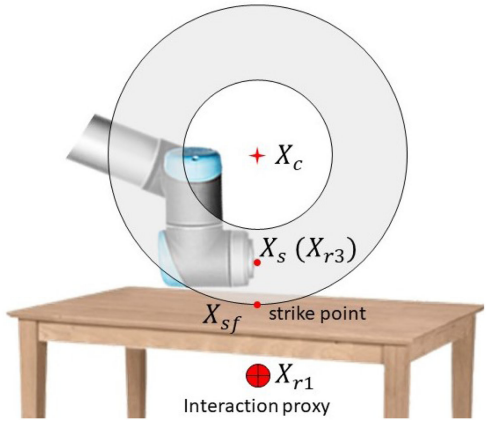


Fig. 5. Changing the center of the workspace due to the presence of a hard contact.

## IV. Controller Design Under State Constraints

According to Section II, the goals of the control architecture are as follows: 1) the robot can smoothly track the interaction proxy without any perturbations; 2) the internal states of the robot (e.g., velocity and force) have their own adaptive boundaries; and 3) the system's stability cannot be affected by potential dynamic disturbances. To reach the above goals, this section utilizes the backstepping method to establish the final control laws.

The unknown dynamics of a robot consisting of multiple DoFs can be expressed as

$$\mathbf{M}_s(q_s)\ddot{q}_s + \mathbf{C}_s(q_s, \dot{q}_s)\dot{q}_s + \mathbf{G}_s(q_s) + \mathbf{B}_s(q_s, \dot{q}_s) = \tau_s + J^T F_e \tag{11}$$

where $q_s$, $\dot{q}_s$, and $\ddot{q}_s$ are vectors of the joint displacement, velocity, and acceleration, respectively. $\mathbf{M}_s(q_s)$ and $\mathbf{C}_s(q_s, \dot{q}_s)$ denote the inertia matrix and the centrifugal and coriolis matrices, respectively. $\mathbf{G}_s(q_s)$ denotes the gravity, and $\mathbf{B}_s(q_s, \dot{q}_s)$ denotes bounded disturbances, including robot frictions. $\mathbf{M}_s(q_s)$, $\mathbf{C}_s(q_s, \dot{q}_s)$, $\mathbf{G}_s(q_s)$, and $\mathbf{B}_s(q_s, \dot{q}_s)$ are unknown. $\tau_s$ is the control torque. $J$ is the Jacobian matrix that satisfies $\dot{x}_s = J\dot{q}_s$; therefore, $\ddot{x}_s = J\ddot{q}_s + \dot{J}\dot{q}_s$. $F_e$ is the passive environment force; in this article, we assume that $F_e$ is unknown, and we use the force observation method in [29] to derive the estimated force $F_e^*$, which may contain estimation errors as $F_e = F_e^* + \Delta_e$, where $\Delta_e$ is the estimation error that consists of unknown dynamics, noises, and acceleration information.

Since the above robot dynamics are unknown, we apply the type-2 fuzzy neural network in [15] and [30] to estimate the overall dynamics; this network can attain a high accuracy and is robust against uncertainties. Therefore, the dynamics (11) are expressed by a combination of multiple linear local models as

$$M_s\ddot{q}_s + C_s\dot{q}_s + D_s q_s + E_s + \varpi_s = \tau_s + J^T F_e^* \tag{12}$$

where $M_s$, $C_s$, $D_s$, and $E_s$ are weighted sums of the local models' coefficients with fuzzy membership grades as weights. The fuzzy membership grades are a group of dynamic functions of the system inputs. $\varpi_s$ is the sum of unknown estimation errors, noises, disturbances, and $\Delta_e$. Therefore, $M_s$, $C_s$, $D_s$, and $E_s$ are time-varying coefficients that describe the nonlinear robotic systems in (11).

The sliding surface of the sliding-mode control has a significant influence on the robotic system. With the sliding surface converging to zero in a finite time, the desired system performance can be achieved. In this article, the nonsingular fast terminal sliding mode (NTFSM) is chosen as

$$s_1 = \int (c_1 sig^{\varphi_1}(e_s) + c_2 S(e_s) + c_3 \dot{e}_s) \tag{13}$$

where $\text{sig}^a(x) = [|x_1|^{a_1}\text{sign}(x_1), \ldots, |x_n|^{a_n}\text{sign}(x_n)]^T$. $c_1, c_2$, and $c_3$ are positive constant control gains. $\varphi_1 > 1$ is a constant. Finally, to avoid singularity, $S(e_s)$ is designed as

$$S(e_s) = \begin{cases} \text{sig}^{\varphi_2}(e_s), & \text{if } e_s >= \epsilon_s \\ \text{sig}^2(e_s), & \text{if } e_s < \epsilon_s \end{cases} \tag{14}$$

where $0 < \varphi_2 < 1$ is a positive constant and $0 < \epsilon_s \ll 1$ is a small value close to zero.

When the sliding surface in (13) converges to zero, the convergence time can be derived as

$$T_s = \int_0^{|e_s(0)|} \frac{1}{\frac{c_1}{c_3}e_s^{\varphi_1} + \frac{c_2}{c_3}e_s^{\varphi_2}} de_s \leq \frac{c_3}{c_1}\frac{1}{\varphi_1 - 1} + \frac{c_3}{c_1}\frac{1}{1-\varphi_2}. \tag{15}$$

From (15), the small value of $c_3$ ($c_3 < c_1, c_3 < c_2$) can efficiently shorten the convergence time. The first and second derivatives of the sliding surface $s_1$ can be written, respectively, as

$$\begin{aligned} \dot{s}_1 &= c_1 \text{sig}^{\varphi_1}(e_s) + c_2 S(e_s) + c_3 \dot{e}_s \\ \ddot{s}_1 &= c_1\varphi_1 |e_s|^{\varphi_1 - 1}\dot{e}_s + c_2\varphi_2 |e_s|^{\varphi_2 - 1}\dot{e}_s + c_3 \ddot{e}_s. \end{aligned} \tag{16}$$

Substituting the fuzzy-based dynamic model in (12) into (16) and letting $\Psi(e_s, \dot{e}_s) = c_1\varphi_1|e_s|^{\varphi_1-1}\dot{e}_s + c_2\varphi_2|e_s|^{\varphi_2-1}\dot{e}_s$, we can derive

$$\begin{aligned} \dot{s}_1 &= s_2 \\ \dot{s}_2 &= s_3 \\ \dot{s}_3 &= \frac{d}{dt}(\Psi(e_s, \dot{e}_s) + c_3(JM_s^{-1}\tau_s + JM_s^{-1}J^T F_e^* - \Omega(q_s, \dot{q}_s) \\ &\quad - \Delta_1 + \dot{J}\dot{q}_s - \ddot{X}_r(t-T))) \end{aligned} \tag{17}$$

where $\Omega(q_s, \dot{q}_s) = JM_s^{-1}C_s\dot{q}_s + JM_s^{-1}D_s q_s + JM_s^{-1}E_s$ are known terms and $\Delta_1 = JM_s^{-1}\varpi_s$ contains unknown noises, disturbances, and the errors of the environmental force estimation.

To find a proper control input for the robotic system in (17) and to simultaneously satisfy the proposed regulations on the position, velocity, and force introduced in Section II, a backstepping design procedure is proposed. The BLF defined in [31] is also applied in the backstepping design to address the state constraints. Based on (17), the following change of coordinates is introduced:

$$z_1 = s_1 - \alpha_1, \quad z_2 = s_2, \quad z_3 = s_3 - \alpha_2 \tag{18}$$

where $\alpha_1$ and $\alpha_2$ are virtual control terms. From (18), we can derive

$$\dot{z}_1 = z_2 - \dot{\alpha}_1 \tag{19}$$

$$\dot{z}_2 = z_3 + \alpha_2. \tag{20}$$

*Step 1:* Define a positive Lyapunov function $V_1$ to be

$$V_1 = \frac{1}{2} z_1^T z_1. \tag{21}$$

Choose the virtual control term $\alpha_1$ to be

$$\dot{\alpha}_1 = k_1 z_1 \tag{22}$$

where $k_1$ is a positive constant gain. By using (19) and (22), the differential of $\dot{V}_1$ can be determined to be

$$\dot{V}_1 = -k_1 z_1^T z_1 + z_1^T z_2. \tag{23}$$

*Step 2:* The estimated environmental force is required not to exceed the constant bound $B_f$, which is defined as a hard contact. Accordingly, the stiffness of the environmental force that corresponds to the robot's position is the main information needed. For simplicity of the controller design, we use (24) as a simple expression of $F_e^*$:

$$F_e^* = -B_e s_2 \tag{24}$$

where $B_e$ is a diagonal matrix that denotes the stiffness of the estimated environmental force. The differential of $F_e^*$ can be expressed as $\dot{F}_e^* = -B_e\dot{s}_2$. Substituting the estimated environmental force into the BLF, we consider the following Lyapunov function (25) with the condition of $|F_e^*(i)| < B_f(i)$ for future controller design:

$$V_2 = V_1 + \frac{1}{2}\sum_{i=1}^{6} \log \frac{B_f(i)^2}{B_f(i)^2 - F_e^*(i)^2}. \tag{25}$$

Based on (20)–(24), the differential of $V_2$ can be derived as

$$\begin{aligned} \dot{V}_2 &= -k_1 z_1^T z_1 + z_1^T z_2 + \sum_{i=1}^{6} \frac{B_e(i,i)^2 z_2(i)\dot{z}_2(i)}{B_f(i)^2 - F_e^*(i)^2} \\ &= -k_1 z_1^T z_1 + z_1^T z_2 + z_2^T H z_3 + z_2^T H \alpha_2 \end{aligned} \tag{26}$$

where $H = \text{diag}([\frac{B_e(1,1)^2}{B_f(1)^2 - F_e^*(1)^2}, \ldots, \frac{B_e(6,6)^2}{B_f(6)^2 - F_e^*(6)^2}])$. By defining the virtual control term to be $\alpha_2 = -H^{-1}z_1 + \alpha_3$, where $\alpha_3$ is to be introduced later, $\dot{V}_2$ can be simplified as

$$\dot{V}_2 = -k_1 z_1^T z_1 + z_2^T H z_3 + z_2^T H \alpha_3. \tag{27}$$

*Step 3:* The robot's velocity $\dot{X}_s$ needs to satisfy $B_v > \dot{X}_s$, where $B_v = B_{v1}$ or $B_v = B_{v2}$ is a varying constraint. Based on (16) and (18), $B_v > \dot{X}_s$ can also be rewritten as $\beta_v > z_2$, where $\beta_v$ is defined as

$$\begin{aligned} \beta_v &= c_1\varphi_1|e_s|^{\varphi_1 - 1}(B_v - \dot{X}_r(t-T)) \\ &\quad + c_2\varphi_2|e_s|^{\varphi_2 - 1}(B_v - \dot{X}_r(t-T)) + c_3\ddot{e}_s. \end{aligned} \tag{28}$$

Considering the velocity constraint, the following Lyapunov function is applied:

$$V_3 = V_2 + \frac{1}{2}\sum_{i=1}^{6}\log\frac{\beta_v(i)^2}{\beta_v(i)^2 - z_3(i)^2}. \tag{29}$$

Setting $\eta_s = z_3/\beta_v$, the differential of $V_3$ is derived as

$$\begin{aligned}\dot{V}_3 &= \dot{V}_2 + \sum_{i=1}^{6}\frac{1}{1-\eta_s^2(i)}\cdot\left(\frac{z_2(i)\dot{z}_2(i)}{\beta_s^2(i)} - \frac{\dot{\beta}_s(i)z_2^2(i)}{\beta_s^3(i)}\right)\\ &= -k_1 z_1^T z_1 + z_2^T H z_3 + z_2^T H\alpha_3 + z_3^T\theta\dot{z}_3 + z_3^T\lambda z_3 \end{aligned}\tag{30}$$

where $\theta = \text{diag}([\frac{1}{(1-\eta_s^2(1))\beta_v^2(1)},\ldots,\frac{1}{(1-\eta_s^2(6))\beta_v^2(6)}])$ and $\lambda = \text{diag}([\frac{-\dot{\beta}_s(1)}{(1-\eta_s^2(1))\beta_v^3(1)},\ldots,\frac{-\dot{\beta}_s(6)}{(1-\eta_s^2(6))\beta_v^3(6)}])$. According to (17), by substituting the robot's dynamics into (30), $\dot{V}_3$ can be further expressed as

$$\begin{aligned}\dot{V}_3 = &-k_1 z_1^T z_1 + z_2^T H z_3 + z_2^T H\alpha_3 + z_3^T\lambda z_3\\ &+ z_3^T\theta\left(\frac{d}{dt}(\Psi(e_s,\dot{e}_s) + c_3(JM_s^{-1}\tau_s + JM_s^{-1}J^T F_e^*\right.\\ &\left.- \Omega(q_s,\dot{q}_s) - \Delta_1 + \dot{J}\dot{q}_s - \ddot{X}_r(t-T))) - \dot{\alpha}_2\right).\end{aligned}\tag{31}$$

Setting the control input $\tau_s$ to be $\tau_s = \tau_1 + \tau_2 + \tau_3$, $\tau_1$ and $\tau_2$ are

$$\begin{aligned}\tau_1 &= J^{-1}M_s(\Omega(q_s,\dot{q}_s) - \dot{J}\dot{q}_s + \ddot{X}_r(t-T) + \hat{\Delta}_1) - J^T F_e^*\\ \tau_2 &= -J^{-1}M_s c_3^{-1}\Psi(e_s,\dot{e}_s)\end{aligned}\tag{32}$$

where $\hat{\Delta}_1$ is an uncertainty estimator used to compensate for the side effect of $\Delta_1$. By setting $\alpha_3 = -k_2\theta^{-1}H^{-1}z_2$, where $k_2$ is a positive constant gain, the virtual control term $\alpha_2$ can finally be written as

$$\alpha_2 = -H^{-1}z_1 - k_2\theta^{-1}H^{-1}z_2. \tag{33}$$

Substitute (32) and (33) into (31). Compensating for $\Delta_1$, $\dot{V}_3$ can be simplified as

$$\begin{aligned}\dot{V}_3 \le &-k_1 z_1^T z_1 - z_2^T\left(\theta^{-1}k_2 - \frac{H}{2}\right)z_2\\ &+ z_3^T\theta\left(\frac{d}{dt}(c_3 JM_s^{-1}\tau_3 - \alpha_2) + \theta^{-1}\lambda z_3 + \frac{1}{2}\theta^{-1}H z_3\right).\end{aligned}\tag{34}$$

We set $\tau_3$ to be

$$\tau_3 = J^{-1}M_s\left(\frac{\alpha_2}{c_3} - c_3^{-1}\int k_3 z_3\right) \tag{35}$$

where $k_3$ is a positive constant gain. Under the situation that the uncertainties are compensated for, $\dot{V}_3$ can finally be written as

$$\dot{V}_3 \le -k_1 z_1^T z_1 - z_2^T\left(\theta^{-1}k_2 - \frac{H}{2}\right)z_2 - z_3^T(\theta k_3 - \lambda - H)z_3 \tag{36}$$

where $\theta^{-1}k_2 \ge \frac{H}{2}$ and $\theta k_3 \ge \lambda + \frac{H}{2}$. Then, $\dot{V}_3$ is negative semidefinite, which implies that the terms $z_1, z_2$, and $z_3$ will converge to zero in a finite-time interval. Therefore, it can be concluded that the proposed backstepping NTFSM system is asymptotically stable.

To design the uncertainty estimator, the fuzzy-based system dynamics in (12) can be rearranged as

$$\begin{cases}\dot{\kappa}_1 = \kappa_2\\ \dot{\kappa}_2 = JM_s^{-1}\tau_s + JM_s^{-1}J^T F_e^* - \Omega(q,\dot{q}) + \dot{J}\dot{q}_s - \kappa_3\\ \dot{\kappa}_3 = \omega\end{cases} \tag{37}$$

where $\kappa_1 = X_s$, $\kappa_2 = \dot{X}_s$, and $\kappa_3 = \Delta_1$ are the states in (37) and $\omega$ is the differential of the unknown uncertainty.

Based on the dynamic state function in (37), we propose the following uncertainty estimator:

$$\begin{cases}\dot{Y}_1 = Y_2 - d_1\varepsilon_1\\ \dot{Y}_2 = JM_s^{-1}\tau_s + JM_s^{-1}J^T F_e^* - \Omega(q,\dot{q}) + \dot{J}\dot{q}_s - Y_3\\ -d_2\varepsilon_2 - d_3\varepsilon_1\\ \dot{Y}_3 = -d_4\text{sig}^{\varphi_2}(\varepsilon_1)\end{cases} \tag{38}$$

where $d_{1-4}$ are constant parameters. $\varepsilon_{1-3}$ are defined as

$$\varepsilon_1 = Y_1 - \kappa_1, \quad \varepsilon_2 = Y_2 - \kappa_2, \quad \varepsilon_3 = Y_3 - \kappa_3. \tag{39}$$

From (37) and (38), we can derive

$$\begin{cases}\dot{\varepsilon}_1 = \varepsilon_2 - d_1\varepsilon_1\\ \dot{\varepsilon}_2 = -\varepsilon_3 - d_2\varepsilon_2 - d_3\varepsilon_1\\ \dot{\varepsilon}_3 = -d_4\text{sig}^{\varphi_2}(\varepsilon_1) - \omega.\end{cases} \tag{40}$$

During the steady state with $\dot{\varepsilon}_{1,2,3} = 0$, the state function in (40) can be expressed as

$$\begin{cases}|\varepsilon_2| = d_1|\frac{\omega}{d_4}|^{\frac{1}{\varphi_2}}\\ |\varepsilon_3| = d_2 d_1|\frac{\omega}{d_4}|^{\frac{1}{\varphi_2}} + d_3|\frac{\omega}{d_4}|^{\frac{1}{\varphi_2}}\\ |\varepsilon_1| = |\frac{\omega}{d_4}|^{\frac{1}{\varphi_2}}.\end{cases} \tag{41}$$

From (41), by selecting $d_4$ to be large enough and by selecting $\frac{1}{\varphi_2}$ to ensure that $|\frac{\omega}{d_4}|$ is small enough, $\varepsilon_1$ can be efficiently guaranteed to be small, which makes $\varepsilon_2$ and $\varepsilon_3$ close to zero at the steady state. Therefore, with $Y_1$ and $Y_2$ closely tracking $\kappa_1$ and $\kappa_2$, $Y_3$ can be guaranteed to track the uncertainties. In (32), $\hat{\Delta}_1$ is the output $Y_3$ of (38).

For the control law $\tau = \tau_1 + \tau_2 + \tau_3$, $\tau_1$ is used to strengthen the system's stability by compensating for the system dynamics and uncertainties. $\tau_2$ is actually a velocity damper related to the position errors; at the steady state with the position error $e_s$ close to zero or the velocity error $\dot{e}_s$ close to zero, $\tau_2$ is close to zero. In $\tau_3$, based on (20), (22), and (33), $\alpha_2$ can be written as

$$\alpha_2 = -H^{-1}\frac{k_1}{\mathbb{S}+k_1}\frac{s_2}{k_1} - k_2\theta^{-1}H^{-1}s_2 \tag{42}$$

where $\mathbb{S}$ denotes the Laplace operator. $\frac{k_1}{\mathbb{S}+k_1}$ can be seen as a low-pass filter with a cutoff frequency $k_1$. During free motion ($F_e^* = 0$), $\max(H^{-1})$ is $\text{diag}([\frac{B_f(1)^2}{B_e(1,1)^2},\ldots,\frac{B_f(6)^2}{B_e(6,6)^2}])$.

Substituting (42) into (35), we can see that during free motion, $\theta^{-1}$ can regulate the velocity according to the changing velocity

constraints $B_{v1}$ and $B_{v2}$. When contacting a physical barrier, the term $H^{-1}$ can make $\tau_3$ decrease to zero. By virtue of the spring-like feedback force provided by $-J^T F_e^*$ in $\tau_1$, the robot will move away from the physical barrier. Moreover, the spherical workspace is changed so that the interaction proxy loses its control priority, and the robot can stay in its current position rather than strike the physical barrier again.

## V. Discussion

This article introduces a new MR-based teleoperation system for enhancing the telepresence and maneuverability of a robotic system. This article provides three main contributions as follows.

1) For enhancing a system's telepresence, a novel MR-based interface is designed that effectively combines virtual and real components. From the designed MR interface, the 3-D model of the robot dynamically follows the real robot's state, and the point cloud from the camera provides the operator with full vision of the movement of the robot and the remote environment, which allows the operator to clearly grasp the situation of the environment and the robot's work performance. Previous research has already demonstrated the superiority of VR-based interfaces over conventional displays. For instance, in [32], the authors demonstrate a significant improvement in collaborative task performance when users are provided with stereoscopic vision feedback. Furthermore, a quantitative evaluation of the improvement of a VR-based interface is provided in [28]; this interface provides better situational awareness through HMD-based pose tracking on a teleoperation task. In our system, the interface implements an AV paradigm, in which a virtual scene is augmented by real-time data from the robot's task space. The choice of using an AV-based interface is stipulated by several factors. First, similar to previous MR-based interfaces, in this article, the operators can freely move in the space and view the task space from different angles. Stereoscopic vision and a parallax effect are achieved by using an HMD and head-pose tracking. The interaction of operators with virtual objects in the scene is accomplished using hand-gesture tracking. Second, to decouple the operator from the control loop and thus eliminate some control noise, the operator can interact with only virtual objects, and the operator's gestures are not directly tracked by the robot. Finally, an AV-based interface offers better opportunities for incorporating visual cues for the operator, such as using spot lights in the task space, which opens up possibilities to isolate operators from distractions and direct their attention to the task.
2) For enhancing the system's maneuverability, a new interaction proxy is proposed for human–machine interactions that constitutes a large improvement compared to the systems proposed in [22], [27], [28], and [33]. In these systems, the operators use physical handles to directly control the robot and cannot let go of the handles when performing the tasks. This control mode has three main drawbacks. First, this control mode compels the robot to always track the operator's behavior. Consequently, the operator must be very cautious and cannot freely move during the whole control process, and sometimes he/she has to maintain an uncomfortable gesture to drive the robot for a long time, which makes the teleoperation task a physically and mentally exhausting job. Second, with the physical restrictions of master controllers [27], the system suffers from workspace differences and kinematic redundancy between the master and the slave; thus, the workspace of the slave robot can be seriously restricted, and the possible poses of the robot cannot be fully used. Third, there is no fault tolerance regarding human motion. The inevitable shaking of the operator's body can cause the robot to vibrate, and thus, the robot may fail the overall task. Therefore, these systems are not very user-friendly and do not support long-term human operation. In contrast, the proposed interaction proxy allows the operator to be freely decoupled from the control loop. It does not require the operator to hold anything and offers the operator a large DoF so that he/she can conduct the teleoperation through comfortable human motions without worrying that his/her unintentional movements may damage the robot or cause it to fail the task. Additionally, the operator can freely manage the breaking time when performing a task so that he/she can stop to take a rest or to observe the robot's states and motion to better plan the next step in the teleoperation task. In addition, the interaction proxy has no physical limitations such that it can take full advantage of the robot's overall workspace and pose. Moreover, this interaction proxy provides virtual buttons (e.g., an activate/deactivate button) for the operator to freely use to implement different functions of the robot.
3) To the best of the authors' knowledge, most existing VR-based teleoperation systems do not consider robot state regulations, which are closely related to a system's maneuverability and task achievement (e.g., [22], [24], [25], [28], and [33]). In the design of the control algorithm of this article, two control modes, CMM and FMM, are proposed for coarse movement and fine movement, respectively, of the robot. In these two control modes, different state regulation strategies are designed to improve the robot's movement. The effective combination of these two control modes has the following advantages. First, the CMM can decouple the proxy from the control loop and allow the operator to freely regulate the desired position and orientation of the robot without influencing the robot. In the FMM, the operator needs to control only the position rather than exhaustively paying attention to both the position and the orientation of the robot. Second, when performing fine movements, an untrained operator may move the interaction proxy to a wrong place, which will mislead the robot or damage the environment. The proposed fuzzy-based position regulation method in (1)–(5) has a strong tolerance for this type of operational fault and restricts the robot's motion in an online-updated small workspace. Third, the velocity regulation strategy in (6)–(10) effectively addresses the motion discontinuity

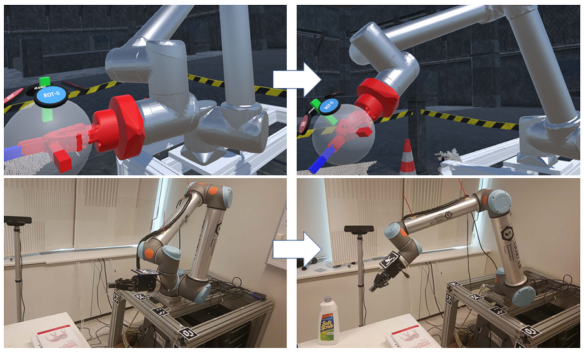

Fig. 6. Experiment for the CMM.

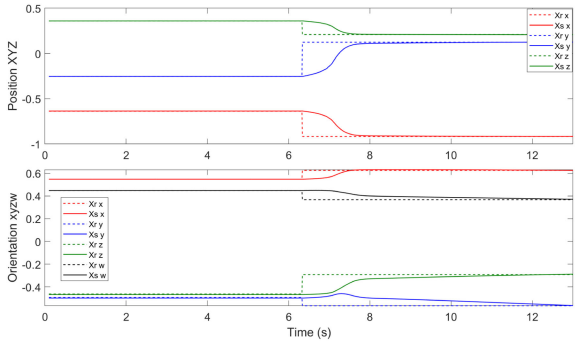


Fig. 7. Position and orientation tracking in the CMM.

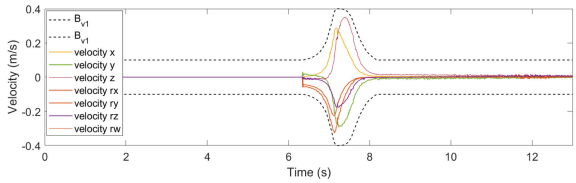


Fig. 8. Velocity regulation in the CMM.

problem caused by a discontinuous reference position from the interaction proxy that offers the robot smooth and safe motions with an adaptive velocity. Fourth, without haptic feedback, the operator cannot control the contact force to the environment of the previous system [22], which can seriously threaten the system's safety. In the proposed system, the force restriction strategy enables the robot to always contact the environment gently; however, the interaction proxy's movement is large. This strategy can guarantee the system's safety during contact.

## VI. Experimental Results

In this section, experimental results are presented to demonstrate the effectiveness of the proposed MR-based teleoperation system. The overall experimental setup includes a Universal Robot (UR10) mounted with a Robotiq-85 industrial gripper, an RGB-D sensor (Asus Xtion Pro), and an HTC Vive MR helmet equipped with a Leap Motion hand gesture sensor. An ROS is used on the slave side as a foundation for the robot's control algorithms. The time delay between the interface side and the UR10 robot is approximately 200 ms. The control gains of the proposed control algorithm are set to $B_e = 0.1$, $\varphi_1 = 2$, $\varphi_2 = 0.8$, $k_1 = 20$, $k_2 = 40$, $k_3 = 20$, $d_1 = 2.8$, $d_2 = 11$, $d_3 = 4.5$, and $d_4 = 7.5$. The gains for the sliding surface $s_1$ are set to $c_1 = 0.5$, $c_2 = 0.5$, and $C_3 = 0.1$.

### A. CMM Test

In the first experiment, we test the position and velocity regulations in the CMM, as shown in Fig. 6. In this experiment, the UR10 robot conducts a relatively long-distance movement. The lower boundary $b_{l1}$ and the upper boundary $b_{u1}$ of the velocity are set to 0.1 and 0.4 m/s, respectively. The upper and lower bounds of the change rate $\sigma_1$ are set to $\sigma_u = 6$ and $\sigma_l = 0.5$, respectively. In the CMM, the radii of the concentric sphere workspaces $R$ and $r$ are larger than the robot's physical workspace ($R = 2.1$ m, $r = 2$ m), so the interaction proxy has full control of the robot. Fig. 7 shows the position and orientation tracking between the robot and the interaction proxy, and Fig. 8 shows the velocity regulation of the robot during a long-distance movement. In the two figures, at 4.2 s, the pose of the interaction proxy suddenly changes. Then, since the velocity boundary is at its lower bound, the robot starts moving at a low velocity. During the movement, the boundary change rate increases from 0.5 to 6, which makes the velocity boundary gradually increase to its upper bound of 0.4 m/s. Thus, the robot can have a relatively high speed during its movement. While attempting to reach the desired pose, the velocity boundary automatically decreases to 0.1 m/s to allow the robot to stably track the proxy to the desired pose. During the overall movement, by using the BLF, the robot's velocity can be efficiently limited by the variable velocity boundary, and the proposed sliding surface can provide the robot with a fast and stable tracking performance.

In comparison, previous studies (see, e.g., [22], [28], and [33]) always couple the robot's motion with the operator's movements and gestures, and the operator has to physically grasp a handle and cannot let go of it when controlling the robot. The inevitable trembling of the operator's body can cause the robot to vibrate, and thus, the robot may fail the overall task; moreover, the operator may have to maintain an uncomfortable pose to control the robot.

### B. FMM Test

In the second experiment, we test the position and velocity regulations in the FMM, as shown in Fig. 9. The FMM is used to provide the robot with continuous tracking during some fine movements. Fig. 10 shows the position regulation, Fig. 11 shows the fuzzy memberships, and Fig. 12 shows the velocity regulation. The upper and lower velocity boundaries are $b_{u2} = 0.8$ m/s and $b_{l2} = 0.3$ m/s, respectively. The radii of the concentric sphere workspaces are $r = 0.2$ m and $R = 0.3$ m, respectively. The change rates $\sigma$ and $\iota$ for the fuzzy memberships in (2) are set to 4 and 200, respectively. During the FMM, the robot's orientation is constant, as it is predetermined by the robot's orientation in the CMM. In these figures, prior to 5 s, the interaction proxy is moved within the small sphere

Fig. 9. Experiment for the FMM.

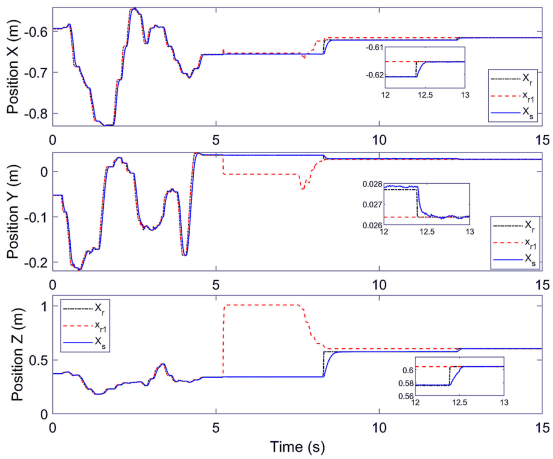


Fig. 10. Position regulation in the FMM.

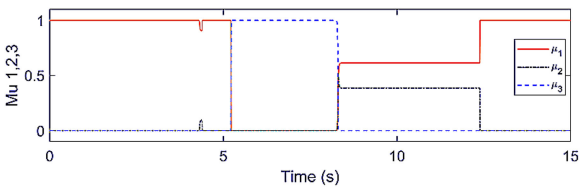


Fig. 11. $\mu_1$, $\mu_2$, and $\mu_3$.

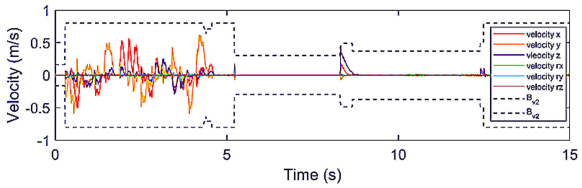


Fig. 12. Velocity regulation in the FMM.

workspace so that it maintains full control of the robot. Additionally, since the velocity boundary is not decreased ($\mu_1$ is always 1), the robot can rapidly and closely track the interaction proxy. Moreover, the workspace continues updating so that the robot position does not have a strict limitation. At 5.3 s, the operator makes an operational fault, in which the interaction proxy is suddenly pushed far away, as shown in Fig. 9. $\mu_3$ immediately changes to 1, which makes the robot stay at its current position rather than continuing to follow the interaction proxy. The fuzzy-based position regulation can efficiently prevent the robot from moving into an unwanted pose, which may cause the robot to fail its task. The velocity boundary also decreases to 0.3 m/s.

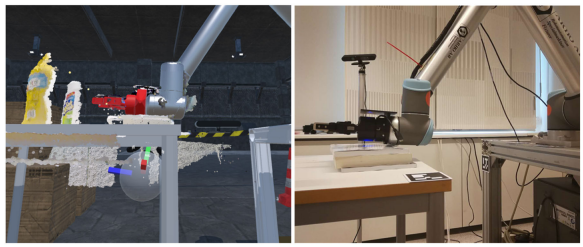

Fig. 13. Experiment for force regulation.

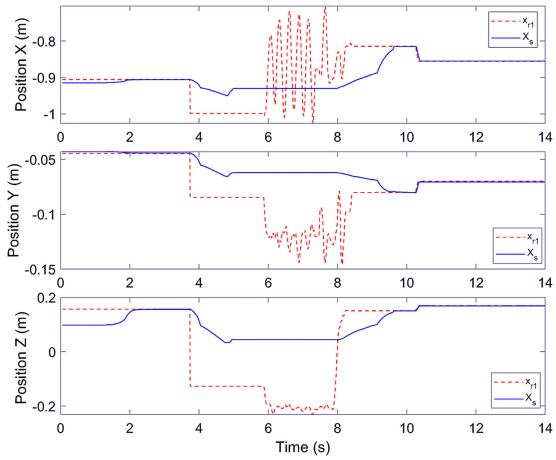


Fig. 14. Position under contact.

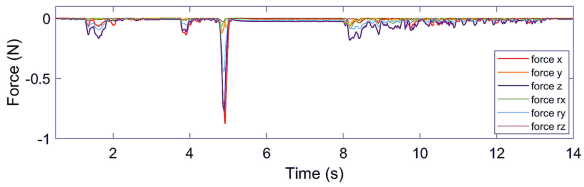


Fig. 15. Force under contact.

When moving the interaction proxy to a reasonable position, the robot gradually approaches and finally tracks the interaction proxy. Note that from 8 to 13 s, the robot tracks $X_r$ rather than the interaction proxy. Then, the robot finally tracks the interaction proxy. This protocol is used to give the operator enough time to adjust the interaction proxy to a desired pose. From 5.3 to 13 s, the velocity is bounded to be low because of the operational fault; this regulation prevents the robot from suddenly jumping to a worse pose. After the robot tracks the interaction proxy to a reasonable pose, the velocity boundary is automatically restored.

In comparison, in [22], [28], and [33], since the robot always tracks the operator's physical handle, if such systems repeat our experiment by quickly moving the handle far away from the robot, the robot will track the handle to the wrong place at an uncontrollable speed, which will threaten the system's safety.

### C. Force Regulation Test

In this experiment, we test the force regulation of the control system, as shown in Fig. 13. The force boundary $B_f$ is set to 1 N. Figs. 14 and 15 show the position and the environmental force, respectively, under the contact motion. After 3.8 s, the

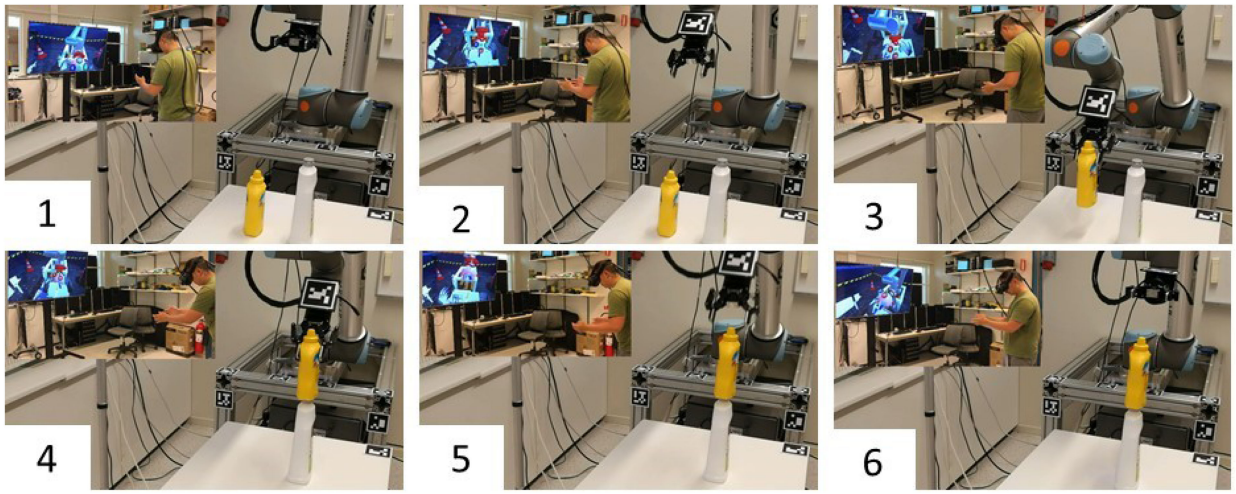


Fig. 16. Real pick-and-place experiment. The yellow bottle is gently placed above the white bottle using the proposed MR-based teleoperation system.

robot follows the position of the interaction proxy and moves down; at 5 s, the robot contacts a physical barrier (a pile of books). Suddenly, at this time, the environmental force is close to the force boundary $B_f$, making $\tau_3$ close to zero. Therefore, the feedback force $-J^T F_e^*$ makes the robot jump up slightly and move away from the physical barrier. Additionally, the workspace is changed accordingly. Therefore, even though the interaction proxy is largely moved under the pile of books from 6 to 8 s, the robot will not follow the interaction proxy, ensuring that it does not contact the physical barrier again. When the interaction proxy is moved above the pile of books (moved back into the workspace), the robot will track the interaction proxy again. The UR10 robot is very sensitive to environmental forces and may apply a very large force that can potentially damage environmental objects, fail the task, or cause damage to itself. This method can restrict the environmental force and prevent the robot from striking the physical barrier again.

Without any force control method, the robots in [22], [28], and [33] will punch rather than gently touching the environment, thereby damaging the robot and causing the system to crash.

### *D. Comparison Using Pick-and-place Experiments*

Fig. 16 shows an example of a pick-and-place experiment using the proposed system. The operator first uses the CMM to regulate the orientation of the UR10 robot to better grasp the task object (the yellow bottle) in the following movements (see parts 1 and 2 in Fig. 16). Then, the FMM is applied to move the robot's end-effector to grasp and lift the object (see part 3 in Fig. 16). In the next step, the yellow bottle is accurately and stably placed above the white bottle (see part 4 in Fig. 16). According to the proposed force regulation, when the yellow bottle contacts the white bottle, the feedback force prevents the gripper from continuing to apply pressure. Hence, the yellow bottle is gently placed rather than "squeezing" the white bottle. After the griper looses its grip on the object, the operator uses the CMM to move the robot back to its original position and orientation (see parts 5 and 6 in Fig. 16).

TABLE I
COMPARISON EXPERIMENTS USING THE TRADITIONAL SYSTEM

| Num | Failure 1 | Failure 2 | $F_e^*$ at Z | Time | S? |
|---|---|---|---|---|---|
| 1 | Yes | No | 0 N | 117 s | No |
| 2 | Yes | No | 0 N | 97 s | No |
| 3 | Yes | No | 1.57 N | 132 s | No |
| 4 | No | Yes | 3.64 N | 91 s | No |
| 5 | Yes | No | 0.53 N | 93 s | No |
| 6 | No | No | 2.02 N | 147 s | Yes |
| 7 | Yes | No | 0 N | 81 s | No |
| 8 | Yes | No | 1.32 N | 105 s | No |
| 9 | No | Yes | 4.85 N | 118 s | No |
| 10 | No | No | 1.5 N | 104 s | Yes |
| 11 | Yes | No | 2.03 N | 124 s | No |
| 12 | No | No | 2.21 N | 159 s | Yes |
| 13 | No | Yes | 5.78 N | 92 s | No |
| 14 | No | Yes | 3.31 N | 136 s | No |
| 15 | Yes | No | 0 N | 101 s | No |
| 16 | No | No | 1.76 N | 88 s | Yes |

A series of experiments repeating the above pick-and-place experiment in Fig. 16 are conducted to quantitatively compare our proposed system with the traditional system in [28]. In the system of [28], the position and orientation of the robot are fully controlled by a tool physically handled by the operator without a force control algorithm. Each of the two systems conducts the above pick-and-place experiment for 16 trials.

We define that a successful trial must satisfy the following three conditions.

1) An orientation change is requested. The robot's tool is parallel to the table at the origin and must point downward to grasp the bottle, as shown in parts 1–3 in Fig. 16.
2) The yellow bottle must be placed above the white bottle, and neither bottle can fall down, as shown in parts 5 and 6 in Fig. 16.
3) During contact, the robot must not crash (the UR10 robot is extremely sensitive to environmental forces; hence, a large contact force can easily cause the overall system to crash).

The overall results of the 32 trials are shown in Tables I and II. In these two tables, we use five parameters to evaluate

TABLE II
COMPARISON EXPERIMENTS USING OUR PROPOSED SYSTEM

| Num | Failure 1 | Failure 2 | $F_e^*$ at Z | Time | S? |
|---|---|---|---|---|---|
| 1 | No | No | 0.84 N | 73 s | Yes |
| 2 | Yes | No | 0 N | 91 s | No |
| 3 | No | No | 0.21 N | 134 s | Yes |
| 4 | No | No | 0.93 N | 84 s | Yes |
| 5 | No | No | 0.53 N | 72 s | Yes |
| 6 | No | No | 0.64 N | 97 s | Yes |
| 7 | No | No | 0.59 N | 69 s | Yes |
| 8 | No | No | 0.22 N | 81 s | Yes |
| 9 | No | No | 0.86 N | 85 s | Yes |
| 10 | No | No | 0.92 N | 77 s | Yes |
| 11 | No | No | 0.78 N | 95 s | Yes |
| 12 | No | No | 0.80 N | 120 s | Yes |
| 13 | No | No | 0.12 N | 79 s | Yes |
| 14 | No | No | 0.70 N | 98 s | Yes |
| 15 | No | No | 0.29 N | 75 s | Yes |
| 16 | No | No | 0.82 N | 78 s | Yes |

the performances of the two systems. "Failure 1" denotes that the bottles fall down. "Failure 2" denotes that the system crashes due to a large contact force when putting the yellow bottle onto the white one. "$F_e^*$ at $Z$" denotes the estimated environmental force along the $Z$-axis when the robot places the yellow bottle onto the white bottle; additionally, since an $F_e^*$ value exceeding 2.5 N can easily cause the system to crash, the value of $F_e^*$ on the $Z$-axis must satisfy $F_e^* < 2.5$ N. "Time" denotes the operation time to perform the pick-and-place task (from part 1 to part 6 in Fig. 16) (for the failed trials, the operation time denotes the period from the beginning to the time at which the failure occurs). Finally, "S?" denotes if the experiment is successful.

By comparing the results in the two tables, the success rate of our system (15/16) is much larger than that of the traditional system (4/16), as demonstrated by the following reasons. First, the operators using the traditional system need to simultaneously control both the position and the orientation of the robot, and the robot's speed does not have an adaptive regulation. As a result, the operator needs to be very skillful to drive the robot, and any bodily shaking may cause "Failure 1" (the bottles fall down). In addition, as the operators have to maintain a specified posture and cannot be decoupled from the control loop during the manipulation of the robot, it is difficult to clearly check the task's status (e.g., check whether the yellow bottle contacts the white bottle when the robot is placing the former). Therefore, the robot can easily drop the bottle rather than placing the bottle, as shown by the trials with 0 N under the "$F_e^*$ at $Z$" column in Table I. In comparison, the operators using our proposed system can switch control modes to regulate the robot's pose and are freely decoupled from the control loop to check the task's status. Therefore, the object becomes easier to pick and place using our system. Second, in the traditional system without a force regulation, the robot is likely to punch the object with a large force, causing "Failure 2". The operators have to be very careful to guide the robot; otherwise, the overall system will crash. The trials in Table I with a force larger than 2.5 N cause "Failure 2". In comparison, as shown in Table II, with the proposed force regulation, the robot in our system can gently place the object, and the forces are lower than the force boundary (1 N) in all trials. The average operation time of the 15 successful trials using our system is 87.8 s, which is lower than that of the four successful trials using the traditional system (124.5 s). The higher success rate and the lower operation time demonstrate that our system is more effective and user-friendly than its traditional counterpart.

## VII. CONCLUSION

In this article, a new MR-based teleoperation system is proposed to provide immersive visual feedback to the operator. In the proposed MR-based teleoperation system, two control modes are designed to improve the long-distance movement and fine movement of the robot. A series of fuzzy-based algorithms are also proposed to regulate the orientation, position, velocity, and force of the robot to improve the system's maneuverability. Control laws based on a BLF and a backstepping control procedure can guarantee the system's stability under state constraints. With immersive visual feedback and regulated system states, the telepresence of the overall system can be enhanced, which potentially leads to a reduced operational workload and an increased task success rate. Multiple experimental results have been performed to show the feasibility of the proposed system.

## REFERENCES

[1] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.
[2] F. Hashemzadeh, M. Sharifi, and M. Tavakoli, "Nonlinear trilateral teleoperation stability analysis subjected to time-varying delays," *Control Eng. Pract.*, vol. 56, pp. 123–135, 2016.
[3] H. Tugal, J. Carrasco, P. Falcon, and A. Barreiro, "Stability analysis of bilateral teleoperation with bounded and monotone environments via Zames–Falb multipliers," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 4, pp. 1331–1344, Jul. 2017.
[4] R. Baranitha, R. Rakkiyappan, R. Mohajerpoor, and S. Al-Wais, "Stability analysis of nonlinear telerobotic systems with time-varying communication channel delays using general integral inequalities," *Inf. Sci.*, vol. 465, pp. 353–372, 2018.
[5] D. Sun, Q. Liao, and H. Ren, "Type-2 fuzzy logic based time-delayed shared control in online-switching tele-operated and autonomous systems," *Robot. Auton. Syst.*, vol. 101, pp. 138–152, 2018.
[6] A. Milstein, T. Ganel, S. Berman, and I. Nisky, "Human-centered transparency of grasping via a robot-assisted minimally invasive surgery system," *IEEE Trans. Human-Mach. Syst.*, vol. 48, no. 4, pp. 349–358, Aug. 2018.
[7] F. Azimifar, S. Ahmadkhosravi Rozi, A. Saleh, and I. Afyouni, "Transparency performance improvement for multi-master multi-slave teleoperation systems with external force estimation," *Trans. Inst. Meas. Control*, vol. 40, pp. 3851–3859, 2018.
[8] D. Sun, F. Naghdy, and H. Du, "Transparent four-channel bilateral control architecture using modified wave variable controllers under time delays," *Robotica*, vol. 34, no. 4, pp. 859–875, 2016.
[9] D. Sun, F. Naghdy, and H. Du, "Time domain passivity control of time-delayed bilateral telerobotics with prescribed performance," *Nonlinear Dyn.*, vol. 87, no. 2, pp. 1253–1270, 2017.
[10] A. M. Okamura, "Methods for haptic feedback in teleoperated robot-assisted surgery," *Ind. Robot: An Int. J.*, vol. 31, no. 6, pp. 499–508, 2004.
[11] Z. Lu, P. Huang, and Z. Liu, "Predictive approach for sensorless bimanual teleoperation under random time delays with adaptive fuzzy control," *IEEE Trans. Ind. Electron.*, vol. 65, no. 3, pp. 2439–2448, Mar. 2017.
[12] K. Hashtrudi-Zaad and S. E. Salcudean, "Transparency in time-delayed systems and the effect of local force feedback for transparent teleoperation," *IEEE Trans. Robot. Autom.*, vol. 18, no. 1, pp. 108–114, Feb. 2002.
[13] M. Cenk Ç., D. Feygin and F. Tendick, "A critical study of the mechanical and electrical properties of the phantom haptic interface and improvements for highperformance control," *Presence: Teleoperators Virtual Environ.*, vol. 11, no. 6, pp. 555–568, Dec. 2002.

[14] C. Tzafestas, S. Velanas, and G. Fakiridis, "Adaptive impedance control in haptic teleoperation to improve transparency under time-delay," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 212–219.

[15] D. Sun, Q. Liao, X. Gu, C. Li, and H. Ren, "Multilateral teleoperation with new cooperative structure based on reconfigurable robots and type-2 fuzzy logic," *IEEE Trans. Cybernet.*, vol. 49, no. 8, pp. 2845–2859, Aug. 2019.

[16] L. Chan, F. Naghdy, and D. Stirling, "Position and force tracking for non-linear haptic telemanipulator under varying delays with an improved extended active observer," *Robot. Auton. Syst.*, vol. 75, pp. 145–160, 2016.

[17] Z. Lu, P. Huang, Z. Liu, and H. Chen, "Fuzzy observer-based hybrid force/position control design for a multiple-sampling-rate bimanual teleoperation system," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 7, pp. 1383–1396, Jul. 2019.

[18] V. Nitsch and B. Farber, "A meta-analysis of the effects of haptic interfaces on task performance with teleoperation systems," *IEEE Trans. Haptics*, vol. 6, no. 4, pp. 387–398, Oct.–Dec. 2013.

[19] P. Huang, P. Dai, Z. Lu, and Z. Liu, "Asymmetric wave variable compensation method in dual-master-dual-slave multilateral teleoperation system," *Mechatronics*, vol. 49, pp. 1–10, 2018.

[20] K. Goldberg, M. Mascha, S. Gentner, N. Rothenberg, C. Sutter, and J. Wiegley, "Desktop teleoperation via the world wide web," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1995, vol. 1, pp. 654–659.

[21] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino, "Augmented reality: A class of displays on the reality-virtuality continuum," *Proc. SPIE*, vol. 2351, pp. 282–293, 1995.

[22] F. Brizzi, L. Peppoloni, A. Graziano, E. Di Stefano, C. A. Avizzano, and E. Ruffaldi, "Effects of augmented reality on the performance of teleoperated industrial assembly tasks in a robotic embodiment," *IEEE Trans. Human-Mach. Syst.*, vol. 48, no. 2, pp. 197–206, Apr. 2018.

[23] H. A. Yanco, J. L. Drury, and J. Scholtz, "Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition," *Human-Comput. Interact.*, vol. 19, no. 1, pp. 117–149, 2004.

[24] C. W. Nielsen, M. A. Goodrich, and R. W. Ricks, "Ecological interfaces for improving mobile robot teleoperation," *IEEE Trans. Robot.*, vol. 23, no. 5, pp. 927–941, Oct. 2007.

[25] H. Martins and R. Ventura, "Immersive 3-D teleoperation of a search and rescue robot using a head-mounted display," in *Proc. IEEE Conf. Emerg. Technol. Factory Autom.*, 2009, pp. 1–8.

[26] F. Abi-Farraj, C. Pacchierotti, O. Arenz, G. Neumann, and P. R. Giordano, "A haptic shared-control architecture for guided multi-target robotic grasping," *IEEE Trans. Haptics*, 2019, doi: 10.1109/TOH.2019.2913643.

[27] D. Ni, A. Y. C. Nee, S. K. Ong, H. Li, C. Zhu, and A. Song, "Point cloud augmented virtual reality environment with haptic constraints for teleoperation," *Trans. Inst. Meas. Control*, vol. 40, pp. 4091–4104, 2018.

[28] D. Whitney, E. Rosen, E. Phillips, G. Konidaris, and S. Tellex, "Comparing robot grasping teleoperation across desktop and virtual reality with ROS reality," in *Proc. Int. Symp. Robot. Res.*, 2017, pp. 1–16.

[29] D. Sun, Q. Liao, T. Stoyanov, A. Kiselev, and A. Loutfi, "Bilateral telerobotic system using type-2 fuzzy neural network based moving horizon estimation force observer for enhancement of environmental force compliance and human perception," *Automatica*, vol. 106, pp. 358–373, 2019.

[30] Q.-F. Liao and D. Sun, "Interaction measures for control configuration selection based on interval type-2 Takagi–Sugeno fuzzy model," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 5, pp. 2510–2523, Oct. 2018.

[31] B. Ren, S. S. Ge, K. P. Tee, and T. H. Lee, "Adaptive neural control for output feedback nonlinear systems using a barrier Lyapunov function," *IEEE Trans. Neural Netw.*, vol. 21, no. 8, pp. 1339–1345, Aug. 2010.

[32] O. Liu, D. Rakita, B. Mutlu, and M. Gleicher, "Understanding human-robot interaction in virtual reality," in *Proc. 26th IEEE Int. Symp. Robot Human Interact. Commun.*, Aug. 2017, pp. 751–757.

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