Towards Applied Remapped Physical-Virtual Interfaces: Synchronization Methods for Resolving Control State Conflicts

Brandon J. Matthews
brandon.matthews@mymail.unisa.edu.au
University of South Australia
Adelaide, Australia

Bruce H. Thomas
bruce.thomas@unisa.edu.au
University of South Australia
Adelaide, Australia

G. Stewart Von Itzstein
gsa@vonitzstein.com
University of South Australia
Adelaide, Australia

Ross T. Smith
ross.smith@unisa.edu.au
University of South Australia
Adelaide, Australia

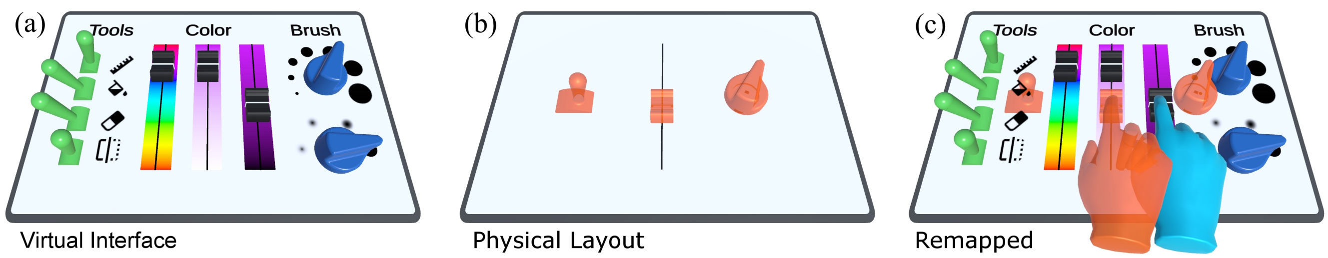


Figure 1: An example of a remapped physical-virtual interface application that would require stateful controls. The virtual interface (a) is an interface for a VR drawing application. Positions of a limited set of physical controls with a fixed layout on a passive haptic interface are shown in orange (b). One of the three sliders is remapped to the only available physical slider (c). The user's virtual hand is represented in blue, while their physical hand position is represented in orange.

ABSTRACT

User interfaces in virtual reality enable diverse interactions within the virtual world, though they typically lack the haptic cues provided by physical interface controls. Haptic retargeting enables flexible mapping between dynamic virtual interfaces and physical controls to provide real haptic feedback. This investigation aims to extend these remapped interfaces to support more diverse control types. Many interfaces incorporate sliders, switches, and knobs. These controls hold fixed states between interactions creating potential conflicts where a virtual control has a different state from the physical control. This paper presents two methods, "manual" and "automatic", for synchronizing physical and virtual control states and explores the effects of these methods on the usability of remapped interfaces. Results showed that interfaces without retargeting were the ideal configuration, but they lack the flexibility that remapped interfaces provide. Automatic synchronization was faster and more usable; however, manual synchronization is suitable for a broader range of physical interfaces.

CHI '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3580723

CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**; **Haptic devices**; • **Computing methodologies** → *Perception.*

KEYWORDS

haptic retargeting, interaction, remapped interfaces, user interfaces, virtual reality

ACM Reference Format:
Brandon J. Matthews, Bruce H. Thomas, G. Stewart Von Itzstein, and Ross T. Smith. 2023. Towards Applied Remapped Physical-Virtual Interfaces: Synchronization Methods for Resolving Control State Conflicts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3544548.3580723

1 INTRODUCTION

This paper presents techniques and hardware prototypes that support the creation of rich remapped physical-virtual interfaces for Virtual Reality (VR) through the integration of haptic retargeting and physical input controls. In particular, the presented system and methods are focused on supporting the remapping of stateful input controls, defined as input controls that retain a fixed state based on the user's last input.

Interfaces in VR are often integrated into the virtual world. This might be a simulated cockpit of an aircraft [33, 35], a control panel in a factory digital twin, or simple buttons and light switches around a virtual world. They are also used to control aspects of virtual experiences, such as menus for changing settings, like a filtering system for data visualizations, or choosing from a palette of different painting or modeling tools. Integrating these interfaces with haptic retargeting enables the creation of flexible remapped physical-virtual interfaces in VR that provide real haptic feedback.

In an ideal remapped physical-virtual interface, a user could naturally reach for any control and receive realistic haptic feedback. The virtual interface could be completely reconfigured, and the user could again operate any control they wish. Previous research on remapped physical-virtual interfaces has been limited to relatively simple interactions with only push-buttons. This paper aims to extend this concept by integrating more complex physical interface components such as knobs, switches, and sliders. These controls can support more complex interactions, expanding the potential use cases and applications.

One could imagine a virtual reality experience such as interacting with modeling or drawing applications with many different tools and parameters. Tools could be selected using tactile buttons on a remapped physical-virtual interface. Depending on the chosen tool, a different interface would appear using tangible controls to adjust the parameters. For example, on an interface like the one shown in Fig. 1, a brush color could be selected with three sliders for Hue, Saturation, and Brightness. A knob could configure the size of a brush and the hardness of the brush edge, while switches could be used to toggle on and off an eraser, rulers, mirroring, or fill tool.

These components require additional techniques to support their integration with a remapped interface. They have properties that enable them to hold physical state information. For example, once a slider is moved to a location, the slider stays in that location until moved by the user or under computer control. In Fig. 1 two of the three virtual sliders are in a different state to the single physical slider. Considering a scenario where multiple virtual components are remapped to a single physical slider, each virtual slider could be in a different state from the physical slider. The state of any slider the user wishes to interact with must be synchronized with the physical slider to ensure natural interactions.

This paper explores two approaches for resolving these state conflicts: 1) *automatic synchronization*, where a custom electronically controllable interface automatically adjusts the physical control state to match the virtual state, and 2) *manual synchronization*, where the user must correct the state mismatch, but any standard or commercially available physical interface could be used as the input. These are evaluated in a user study to understand their usability and impact on interaction times. In addition, the retargeted interface is compared against an aligned interface where the physical and virtual layouts are identical to explore the effect of the haptic retargeting illusion on the usability and task performance of interfaces with passive haptic feedback. To emulate a more natural interaction paradigm, users could select their interaction target using a head pointer selection method. This is intended to support typical user interface interactions where no pre-determined order is known.

The study highlighted that the aligned interface performed the best and was most preferred over a remapped interface. The automatic synchronization method performed better than the manual method regarding errors and interaction time.

The primary contributions of this paper are:

- Two proposed methods ("automatic" and "manual") for synchronizing stateful input controls to support a rich remapped physical-virtual interface system.
- A user evaluation of introducing haptic retargeting to user interfaces in virtual reality and the proposed state synchronization methods.

Following this introduction, we examine related works to motivate and place our contribution. Design parameters for a set of control components are discussed, highlighting how each operates uniquely in terms of state information and their manual and automatic synchronization methods. A study follows this to evaluate the two synchronization methods and the usability of remapped interfaces. The results of this study and the future directions of this system are discussed. Finally, we propose some conceptual applications for the future use of remapped physical-virtual interfaces with stateful controls followed by concluding remarks.

2 RELATED WORK

This research builds upon prior and related work in haptic user interfaces in virtual reality and haptic retargeting.

2.1 Haptic User Interfaces in Virtual Reality

The control components of physical interfaces provide affordances and constraints for their operation [34]; for example, the slider has a track it follows, affording a one-dimensional change in a value and constraining the user to linear movement, along with a physical stop at each end that implies a valid range of values. Similarly, users can feel when physical buttons actuate and have been shown to provide higher typing speeds [38], interaction accuracy [24], and throughput [37]. Providing such haptic feedback in VR presents a challenge as these virtual user interfaces are generally dynamic and adjust to the context of the virtual world.

A dynamic physical user interface should provide tangible control components with physical constraints and affordances. The position, orientation, and the number of these components should be adjustable using a computer or other control system. Outside of virtual reality, there has been a considerable body of work exploring the creation of dynamic tangible interfaces. Methods have included physically shape-changing interfaces using pneumatics [18] or mechatronics [13] and user reconfigurable interfaces such as tangible controls for touch screens [20, 42] and spatial augmented reality [40].

VR can provide visually dynamic and spatial user interfaces with similar affordances and constraints to a physical interface. These interfaces typically lack the haptic feedback of physical control components, and the physical hand can pass through the interface breaking the user's feeling of presence and decreasing interaction performance [8]. Previous work explored input approximations for interacting with VR control panels with sliders, buttons, toggle switches, and knobs [41]. This research found that approximated hand interactions where the user interacts without haptic feedback were not generally preferred over controller-based approximations. However, the lack of haptic feedback could be a contributing factor.

Previous research has explored the use of robotics to move physical control components into the same spatial position as a virtual control component. The Snake Charmer, for example, is a robotic arm that can move simple objects or control components into position such that the user touches it at the same time as a virtual equivalent[2]. The main limitation of such a device is speed and range of movement; they can not simulate virtual objects beyond their range and take time to move to the target position. The Haptic Revolver is a similar, handheld haptic controller which can place small components such as tactile buttons and switches under the user's fingertip [43]. As the user touches virtual controls, the wheel rotates to move physical controls in place.

Lindeman et al. found that passive haptic paddles significantly improved user performance for user interface manipulation tasks in VR [25–27]. The user's finger touches the passive haptic paddle when they operate the interface providing some approximated haptic feedback and preventing the hand from passing through the virtual interface. This passive haptic method lacks the feedback from grasping a knob and the physical constraints of moving a slider within a track. An alternative method of providing haptic feedback for dynamic interfaces in VR is the haptic retargeting visuo-haptic illusion.

2.2 Haptic Retargeting for User Interfaces

Visuo-haptic illusions leverage the dominance of visual perception over proprioception, and haptic perception [15]. In VR, these illusions have been applied to create a variety of haptic illusions [1, 6, 47], and also to enable the manipulation of the user's perceived hand position during reaching actions, typically known as hand redirection [45]. To achieve this, an angular and translational gain are applied to the movement of the user's hand. These gains have been shown to remain undetectable [45] up to 4.5° angular gain in either direction and a translational gain of 0.88 to 1.07. This redirection has also been tolerable up to significantly larger amounts of redirection up to 40° angular gain[9]. This range can also be increased using change blindness, combining continuous and instantaneous redirection applied during a user's blinks [44]. Kohli et al. found that users can adapt to the redirection when interacting with disjoint physical-virtual surfaces [23], and the rate of errors while experiencing the illusion is no worse than without the illusion [22].

Haptic retargeting is a type of hand redirection tailored explicitly to reusing passive haptics. In the body warping technique, angular and translational gains are applied to the user's hand such that the user can touch one physical object while believing they are touching one of many multiple virtual objects [5, 9, 17]. Similar illusions have been applied in conjunction with encounter-type haptics [16] and combined with redirected walking to expand the re-usability of haptics.

In previous work, we first applied haptic retargeting to user interface interactions where a single button on a handheld controller was extended to nine virtual buttons [31]. Later, Feick et al. explored visuo-haptic illusions that can extend linear translation and presented a system to support multiple virtual sliders of different lengths using haptic retargeting and one motorized physical slider [12]. This paper further develops these concepts to support more comprehensive user interfaces, providing haptic feedback for buttons, sliders, knobs, and toggle switches in a single integrated system.

Perhaps the most significant limitation of the haptic retargeting technique is the need for the system to understand the user's intent. In the context of remapped physical-virtual interfaces, the system must know, for example, which button the user wants to press before they press it. In an ideal remapped interface system, users should be free to interact with the remapped interface naturally without a scripted interaction order - to achieve this, a selection method is required.

Cheng et al. combine eye gaze and hand trajectory predictions to correctly identify the user's target with high accuracy at 97.5% [9]. However, target size may affect the accuracy, and their haptic targets were quite large compared to typical interface controls. Similarly. Clarence et al. used Long-Short Term Neural Networks for target prediction to achieve 81.1% prediction accuracy within approximately 65% of the reaching movement, allowing for 35% of the movement to perform the haptic retargeting. Matthews and Smith developed a remapped interface using head gaze for target selection, enabling the user to directly control which virtual button they interact with [29] though this requires the introduction of unnatural head movement.

Operation of different control types requires different grasping and pointing hand poses. Feick et al. also found that hand redirection illusions like haptic retargeting are effective regardless of hand pose when grasping [12]. In our implementation and user study, we use Shape Aware Haptic Retargeting [30] to ensure the interactions are accurate, regardless of hand pose.

3 SYNCHRONIZATION METHODS FOR STATEFUL CONTROLS

Remapping between stateful physical and virtual controls, such as switches, sliders, and knobs, can create a state conflict between the physical and virtual representations. To handle this state conflict we propose two synchronization methods: *automatic* using motorized controls and *manual*, where the user corrects the state. Fig. 2 shows the motorized controls and the visualizations used to guide the manual synchronization. To support this in our approach, each virtual component retains knowledge of its internal state, independent of any physical component. Upon identification of the virtual interaction target, the system selects the nearest compatible physical target. If a state conflict exists, either the automatic or manual state synchronization occurs. Once the state matches, haptic retargeting begins, and the user can complete their interaction. The approach used in our system is similar to the technique proposed by Feick et al. to map virtual sliders of different lengths to a single physical slider [12]. Their approach maintains a log of previous virtual states and matches the physical control to the corresponding virtual state. Our approach enables the flexible addition and removal of virtual controls in previously unknown states, the mapping of multiple virtual controls to multiple physical controls, and dynamic or ranked mapping techniques [29].

In the *automatic* synchronization method, electro-mechanical components are used to create controls that can move to any given state without physical user input. The physical devices are shown in Fig. 2a-c. The introduction of electro-mechanical controls converts

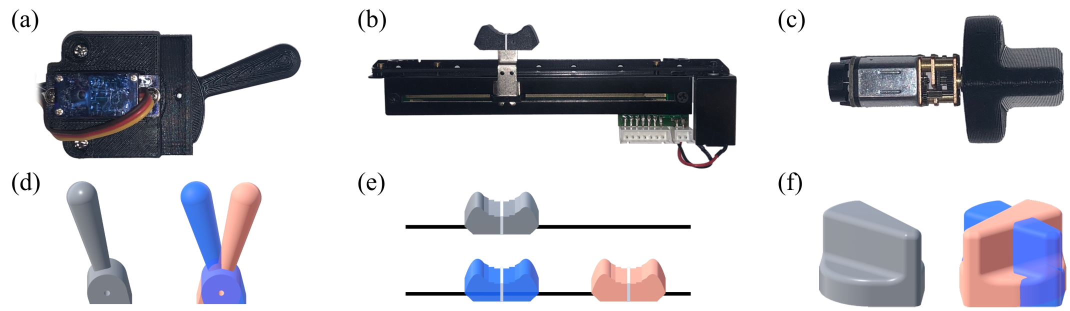


Figure 2: The corresponding physical components used to support the automatic synchronization for the switch, slider, and knob (a, b and c) as well as the visualizations used for the manual synchronization method for the (d, e, and f).

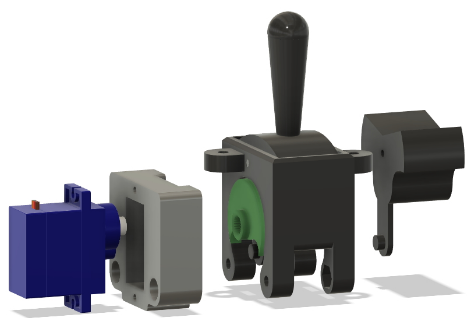


Figure 3: Servo operated Switch with Rocker and Toggle variants. The green component attaches to the servo and interacts with the lever arm on the bottom of the toggle or rocker element.

the passive haptic interface into a dynamic passive haptic [46]. The haptic feedback the controls provides remains passive; however, the controls themselves are dynamic and computer controllable. There is a small period when the component moves to match the virtual state. During this time the user cannot interact with the virtual control.

In the *manual* state synchronization method, the user is shown a ghost of the physical component's real state as shown in Fig. 2d-f. The user must then adjust the ghost to align with the virtual component, where the two become synchronized. Initially, a virtual synchronization method was considered where changes are only made to the virtual interface to account for the state conflict. One approach could be simply changing the virtual control state; however, this could create confusion for the user when a control is not in the expected state. Potentially a linear [12] or rotational gain could be applied to the movement of the slider or knobs such that the available range could be adapted to the virtual slider position; however, in testing, it was found that this approach could not work when the slider and knob were at or near their limits.

3.1 Toggle and Rocker Switch Implementation

Switches come in two common actuation styles, momentary and latching. Momentary switches do not hold their state, while latching switches do. The latching switch is a stateful component and thus can result in a state conflict. Momentary switches have a default state; the switch only enters other states when the user changes and then returns to the previous state itself. For toggle and rocker-type switches, the actuation is different for a latching switch as opposed to a momentary switch. In a latching switch, the component must be actuated in different directions depending on the current state. As a result, the switch must be controllable to allow the system to align the physical control state to the virtual control state.

Actuated switches are rare, so a more accessible, custom servo actuated switch was developed with the ability to control its state automatically. The switch consists of a 3D-printed body with an internal micro-toggle switch that is actuated using a larger 3D-printed toggle or rocker. The concept of toggling a smaller internal switch is based on an open-source switch design from VRFlightSim [1]. To enable automatic actuation, a servo is used to push a lever arm on the 3D-printed toggle or rocker, and the state is read from the internal micro-toggle switch. Fig. 3 shows the exploded render of the design, and the physical prototype of this switch can be seen in Fig. 2a.

This custom servo-actuated switch allows us to emulate a momentary switch in either direction and a latching switch that can change state to match the virtual equivalent. A toggle style switch was used for this implementation, though the device also supports a rocker style switch shown in Fig. 3. To emulate a momentary switch, we first detect when the switch changes to its temporary state, then move the servo to push against the direction of state change back to the default state. This provides the same operation and feedback that off-the-shelf momentary switches provide.

In the manual mode (shown in Fig. 2d), an orange copy of the physical switch appears, aligned with the virtual target, and must be switched to align it with the virtual switch. Once they are in the same state, the physical copy disappears, and the physical and virtual switch states become linked. Fig. 4 outlines the sequence for both the manual and automatic synchronization from the selection of the switch through to their states being properly linked.

[1]Title: 3D Printed Cessna Switch, Source: https://www.thingiverse.com/thing:2794629, CC-BY-NC-ND

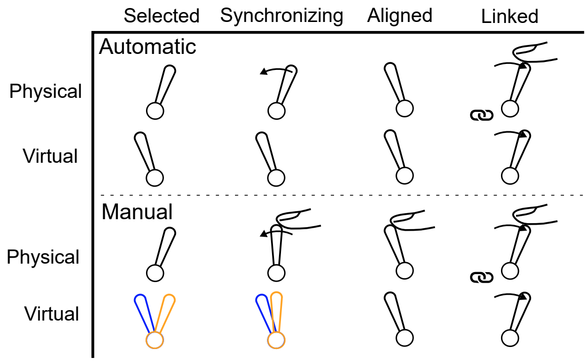


Figure 4: Synchronization sequence for a switch using an automatic rocker/toggle switch (top) and manual synchronization (bottom). In the manual synchronization sequence, the orange control indicates the physical ghost components changing to match the blue virtual control based on the user input.

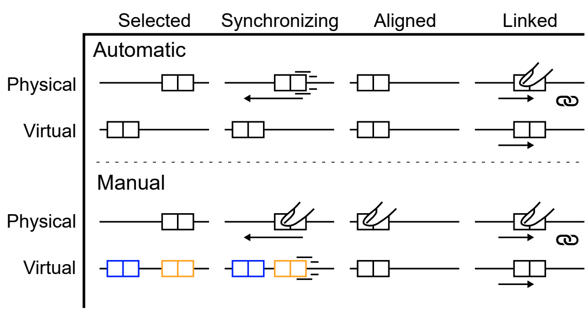


Figure 5: Synchronization sequence for a slider using a motorized slide potentiometer (top) and manual synchronization (bottom). In the manual synchronization sequence, the orange control indicates the physical ghost components changing to match the blue virtual control based on the user input.

3.2 Slider Implementation

Sliders on physical interfaces typically take the form of a slide potentiometer that outputs an absolute position value. For this implementation, the Bourns 10k 10 cm motorized slide potentiometer (PSM01-082A-103B2) shown in Fig. 2b was used for the slider. The slider can be actuated electronically and moved to the same position as the virtual slider when selected. Once the physical slider has matched the position of the virtual slider, they become linked.

In the manual synchronization mode (shown in Fig. 2e), a ghost version of the physical slider is visualized on top of the real version. The user must move the ghost slider to match the virtual slider, at which point the host slider disappears, and the two virtual and physical components become linked. Fig. 5 outlines the sequence for both the manual and automatic synchronization from selecting the slider to their states being correctly linked.

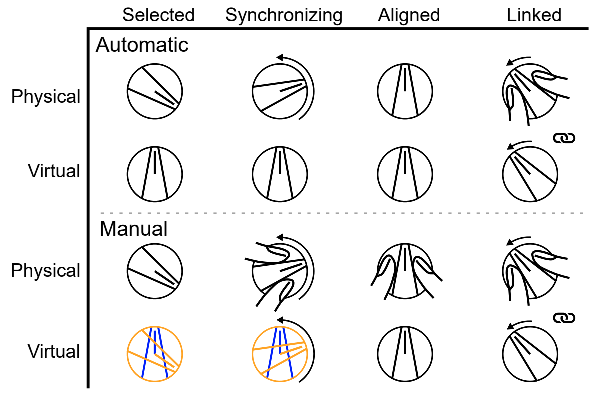


Figure 6: Synchronization sequence for a knob with absolute position using a motorized rotary potentiometer or motor with encoder (top) and manual correct (bottom). In the manual synchronization sequence, the orange control indicates the physical ghost components changing to match the blue virtual control based on the user input.

3.3 Knob Implementation

Knobs on physical interfaces take two forms: limited using rotary potentiometers or continuous using encoders. Continuous rotation knobs generally use rotary encoders to count steps as the knob is rotated. Limited knobs use rotary potentiometers limited to a fixed rotation range and output an absolute position value.

For the automatic synchronization mode, an Alps Alpine linear motorized rotary potentiometer (RK16812MG099) was initially explored as an option for this implementation; however, in testing, it was determined the 12 seconds required to move from the lower limit to the upper limit created too large of a delay as the knob synchronized. Instead, a 6V 30:1 geared motor (FIT0481) shown in Fig. 2c was used along with an encoder to measure its position. The rotation of the knob was limited to 230° by adding physical end stops to the 3D printed mount and knob. Upon connection to the control Arduino, the knobs were zeroed at the starting position, and the upper rotation limit was 1750 steps. The count was then converted to an angle range from 0° to 230°. While the motor feels different from the rotary potentiometer as it rotates, it can move the full range of motion in approximately 1 second.

As with the slider, in the manual synchronization mode (shown in Fig. 2f), a ghost knob appears and must first be aligned with the virtual knob, at which point the ghost knob disappears, and the physical and virtual knobs become linked. Fig. 6 outlines the sequence for both the manual and automatic synchronization from the selection of the slider to their states being correctly linked.

4 EVALUATION OF SYNCHRONIZATION METHODS AND REMAPPED INTERFACES

This evaluation seeks to investigate the usability and task throughput parameters of an aligned physical-virtual interface, compared to a remapped physical-virtual interface with both the automatic and manual synchronization methods to handle state synchronization for remapped interfaces. The study follows a within-participants design with three conditions: Aligned, Auto-Remapped, and Manual-Remapped. In the Aligned condition, the passive haptic interface is identical to the virtual interface. In the Auto-Remapped and Manual-Remapped condition, a remapped physical-virtual interface is used with either the automatic or manual synchronization methods.

4.1 Preliminary Selection Method Evaluation

An inherent factor of the synchronization system is the delay between identifying the virtual target and the corresponding physical component and the subsequent synchronization of the virtual component. To evaluate the synchronization methods as they could be used for a real interface participants should have some freedom in which input control they interact with and when. Furthermore, previous studies of haptic retargeting in the context of user interfaces have used entirely scripted, procedural interactions. Providing interaction freedom enables a practical evaluation of the usability of remapped physical-virtual interfaces.

Previous research has explored target prediction methods for haptic retargeting that identify user targets with high accuracy. They leverage eye gaze and hand trajectory analysis [9] or long-short term neural networks trained on hand motion paths [10] to guess the target as the user reaches toward it. Predictive methods are not suitable for a study where there is a possibility the prediction might be incorrect. Instead, two pointer-based approaches were considered as possible candidates, a head pointer and a hand pointer.

A preliminary study was conducted with 12 participants to evaluate these pointer-based selection methods in the context of haptic retargeting. An ideal selection method would have minimal errors and selection time, be considered usable, and not physically tiring for the user to operate. In the study, participants selected and then pressed one of eight buttons on a remapped interface in a ring with three different diameters (8 cm, 12 cm, and 16 cm) resulting in differing retargeting gains and target proximity. The selection area around each button was a 3 cm x 3 cm square around the button. When the reticle is aimed at a target for a dwell time of 0.3 seconds, the target is then selected.

Hand pointer selection has been widely explored as a method for virtual reality [3, 11] and other displays [21] though there has been little consideration in the context of haptic retargeting. While hand pointer techniques can require additional effort to keep the hand steady while selecting the target [4], we feel this is interesting to investigate as haptic retargeting itself already involves interaction with the hand. Most of the reset methods explored to date rely on retraction of the hand away from the targets [28], and thus they already place the hand in a potentially useful location for target selection .

Similar to the fixed origin technique by Jota et al., in our hand pointer, the ray extends from the user's approximated shoulder position, through a point between the user's thumb and index finger, with a reticle drawn at the point the ray intersects with the interface. The head pointer selection for haptic retargeting is based on our previous implementation and uses a ray in the direction the user is facing [29]. A reticle is drawn at the point the ray intersects a target to visualize the pointer for the user.

Table 1: Additional questions regarding the participant's perception of their hands and the interface.

Ownership	I felt like the hand I saw was my real hand.
Position	I felt like the hand I saw was aligned with my real hand.
Movement	I felt like the movements of the hand I saw represented the movements of my real hand.
Alignment	I felt like the interface I saw matched the interface I touched.
Realism	I felt like I was touching a real interface.

In the study participants generally felt that the hand pointer resulted in more physical demand and required more effort but the difference was not statistically significant. The rate of incorrect selections was similar for both techniques averaging above 99.94% selection accuracy in all layouts. The 16cm diameter layout with a spacing of 6.12 cm between targets had 100% accuracy with zero errors. This accuracy is higher than the previously evaluated predictive methods [9, 10], but naturally, they require additional user control. Given these similarities, the faster performance for interaction and selection times for head pointer led to us selecting the head pointer approach as the selection method for this study. The full details of the pointer implementations, preliminary study design, procedure, and results are provided in Appendix A.

4.2 Measures

This study explores the effect of each stateful synchronization method and haptic retargeting on task performance, task load, and the usability of physical-virtual interfaces. We measured task performance as the task completion time and the number of incorrect interactions performed by the participant. Task completion time was measured as the time between the user selecting the target and completing the task. The number of incorrect interactions was the number of times a user modified the wrong control in each sequence of three tasks. Given the presence of all physical controls, regardless of the virtual control task, users may grab and operate the wrong control. Due to the proximity of the physical controls, participants may not adapt to the retargeting fast enough and could overshoot their target. In this case, they may inadvertently grab and change the wrong control.

In addition to these metrics, participants completed the raw NASA TLX questionnaire [19] to measure task load, and the System Usability Scale [7] to measure the usability of the interface. Additional questions were included regarding the participant's perception of their hands and the interface as listed in Table 1. The response was a 5-point Likert scale from 1:Strongly Disagree to 5:Strongly Agree. After all conditions, participants provided their preferences between the aligned interface, the remapped interface, and automatic and manual synchronization.

4.3 Hypotheses

4.3.1 Task Completion Time. The interface used in the Aligned condition is the ideal configuration with matching and spatially aligned physical and virtual interfaces. As such, it should perform better in all metrics than the retargeted interface with automatic synchronization (Auto-Remapped). As such, it is predicted (H1) that the introduction of haptic retargeting will result in slower movement resulting in increased task completion time.

Manual synchronization introduces an additional interaction step and an increased amount of required changes. Thus it is hypothesized (H2) that the automatic synchronization method will have a shorter task completion time than the manual synchronization method.

- H1 Task completion time will be lower for an aligned passive haptic interface than for a remapped interface.
- H2 Task completion time will be lower for the automatic synchronization method than the manual synchronization method.

4.3.2 Incorrect Interactions. As observed in prior studies of haptic retargeting, it is hypothesized that participants will overshoot or fail to correct for the haptic retargeting. We hypothesize (H3) that without haptic retargeting in the Aligned condition, the number of incorrect interactions will be lower than with haptic retargeting in the Auto-Remapped condition.

The size and selection of physical controls made it difficult to create a uniform layout for every control. The slider control is hypothesized (H4) to produce more incorrect interactions than the knob and switch for two reasons: 1) the slider control is positioned between the other controls and slightly offset to the left, 2) the slider grip position changes, creating more inconsistency in the reach action which could lead to the user being less able to learn the correct action.

- H3 There will be fewer incorrect interactions with an aligned passive haptic interface than with a remapped interface.
- H4 There will be more incorrect interactions for the slider control than the other control types.

4.3.3 Subjective Feedback. For the questionnaire responses, we hypothesize (H5) the interface without haptic retargeting will have better system usability than the Auto-Remapped condition with haptic retargeting. We also hypothesize (H6) that the system usability will be higher for the automatic synchronization method than for the manual method.

For the task load as measured by the NASA TLX questionnaire, we hypothesize (H7) that the mental load, physical load, and effort will be lower for the automatic synchronization method than the manual method. This is due to the additional step added to the tasks and the need for the user to differentiate between the orange physical and blue virtual controls required for the manual method. Furthermore, it is hypothesized (H9) that users will notice more inconsistency between the real and virtual interfaces with the manual synchronization method as they may be more aware of the retargeting when they see the physical control state matches the last control they changed.

The layout of the controls creates a hand redirection gain above the estimated detection thresholds [45]. The maximum possible angular gain with this configuration is 15.2°. This is computed by taking the closest point (P) on the reset plane to the midpoint (M) between the physical control (C) and virtual control (30cm from the midpoint). The angular gain is the angle θ formed by the vectors $\vec{PC}$ and $\vec{PM}$. As a result, the introduction of retargeting for the remapped interface is hypothesized (H8) to increase the reported mental load. Similarly, the participant's perception of the accuracy of their hands and the interface is hypothesized (H10) to be better without haptic retargeting.

- H5 The system will be more usable with an aligned passive haptic interface than a remapped interface.
- H6 The system will be more usable with the automatic synchronization method than the manual synchronization method.
- H7 Participants will experience lower mental load, physical load, and effort with the automatic synchronization method than with the manual synchronization method.
- H8 Participants will experience a lower mental load with an aligned passive haptic interface than a remapped interface.
- H9 Participants will notice more inconsistency between the physical and virtual interfaces with the manual synchronization method than with the Automatic synchronization method.
- H10 Participants will perceive their hands as more accurately aligned with their real hands without haptic retargeting.

4.4 Apparatus

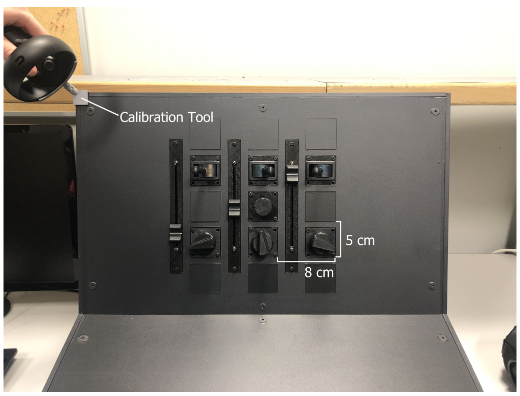
The study experience was developed in the Unity game engine (version 2020.3.30f1). Participants wore an Oculus Quest device connected via the Oculus Link to a high-performance desktop computer running an Intel i7-7820X CPU, 64GB RAM, and an NVIDIA GTX1080Ti GPU. The Oculus Quest was selected as it provides integrated hand tracking that is accurate within 1 cm and more accurate than both the Leap Motion and HTC Vive hand tracking[39].

The physical setup for the study is shown in Fig. 7. The physical interface is a console-style structure with a 15° angle on the top panel. The controls are modular, enabling the interface to support various controls. For the study, a center button was added, with three of each control type (Slider, Knob, and Switch). The Knobs and Switches were placed 8 cm apart, with the Switches 5 cm above the button and the Knobs 5 cm below. Given the size and proportions of the slider control, these were placed between the other controls, also 8 cm apart. The 8 cm spacing was necessary to fit the sliders in between the other controls. The interface and controls were black to minimize interference with near-infrared hand-tracking technologies like the Leap Motion and Oculus Quest.

4.5 Calibration

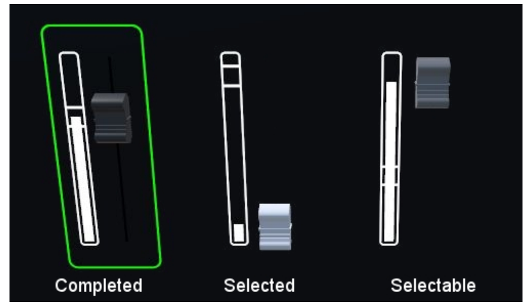
To calibrate the physical interface in the virtual environment, the orientation was determined first using an Oculus Touch controller with a 3D printed attachment, similar to the tool used by Schneider et al. [39] and shown in Fig. 7b. The attachment was aligned with the corner of the interface, and a button was pressed on the controller, moving the interface to match the orientation of the tool. Then hand tracking was enabled, and the participant pressed the button on the interface using their index finger. This accounts for variations in hand size or inaccuracy between the controller and hand tracking. To validate the calibration, the participant was asked to trace their finger around the button and verify the virtual finger followed the edge of the virtual button. The participant could re-calibrate as

(a) Study Room and Setup

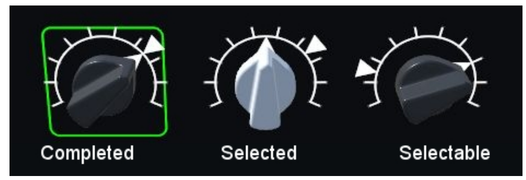


(b) Physical Interface Layout

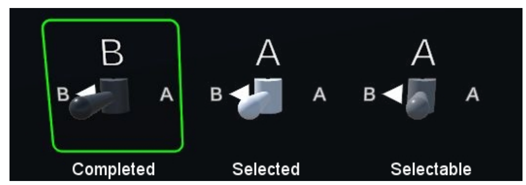
Figure 7: The physical study setup (a) and the interface layout as well as the tool used to calibrate the orientation of the interface (b).



(a) Slider Study Task



(b) Knob Study Task



(c) Switch Study Task

Figure 8: Tasks for each control type as seen by participants. In each image, the left task is completed, the middle task is selected and yet to be completed, and the right task can be selected.

necessary until they stated they could comfortably and accurately press the button.

4.6 Design and Participants

We recruited 24 participants (17 self-reported as male, seven as female, aged 22 to 40 (mean:29.5, σ: 5.524)), with two participants reporting as predominantly left-handed and the other 22 as right-handed. Participants were all required to have normal or corrected to normal vision with no colorblindness issues that would prevent differentiating between Orange and Blue used to visualize the physical and virtual control in the Manual-Remapped condition. Of the participants, one reported using VR systems daily, ten weekly, seven monthly, three yearly, and three reported never using VR. The study followed a within-participants design with each participant experiencing all three conditions. The conditions were fully counterbalanced between the participants with each of the six order permutations experienced by four participants.

4.7 Input Control Tasks

The individual input control tasks are depicted in Fig. 8 and differ depending on the control in question. The participants completed each task (slider, knob, or switch) three times. The order of all task types was randomized for each participant and each condition.

In the **slider task**, the participant is required to move the slider such that the fill bar is within the given area. The area is indicated by two white horizontal lines on a white rectangular scale, and the fill bar is a solid white bar that fills up from the bottom, following the slider. In the **knob task**, the participant is required to rotate the knob such that the outward arrow attached to the knob is within the region indicated but the inward pointing arrow. In the **switch task**, the participant is required to set the switch to the indicated position (A or B), where A is pushed to the right and B is pushed to the right. For the Auto-Remapped and Manual-Remapped conditions, only the center physical control is used for all three tasks. The virtual left and right task controls are remapped to the center physical control using haptic retargeting.

4.8 Haptic Retargeting Method

To achieve the remapping, On-The-Fly haptic retargeting [9] is used to warp the user's virtual hand position and account for the offset between the physical and virtual controls.

This is extended with Shape Aware Haptic Retargeting primitive approximations to ensure the hand is properly aligned with the control upon contact [30]. The knob and switch are each approximated using a 4cm diameter sphere. The slider is approximated by a 3cm diameter, 12cm long capsule that encapsulates the knob and runs the entire length of the slider track. In particular, for the slider control, the primitive approximation ensures the hand is aligned regardless of the slider's state. This allows the warp to be computed based only on the position of the control itself, independent of the state of each control. For example, the participant could accurately touch any point along the track of a remapped slider, regardless of its state.

Between each interaction and while selecting their next control, the participants were required to return their hand to the warp origin. The origin used in this study was a vertical plane placed 30 cm from the center button on the interface based on the approach used by Cheng et al. [9], and our previous static threshold reset technique [28]. The threshold was visualized only on the lower surface of the interface to encourage participants to rest their hands on the interface between tasks. A blue rectangle marked the origin where the edge of the rectangle furthest from the user is aligned with the threshold. The 'reset' was complete anywhere between the user and that edge.

4.9 Procedure

To begin the experiment, participants were provided with a description of the study and completed an informed consent form. Participants were then provided with a PowerPoint presentation to read at their own pace that described in more detail what haptic retargeting is, the task, and the automatic and manual synchronization methods. Before each condition, the interface was calibrated following the procedure outlined in Section 4.5, and the physical interface reconfigured itself to a default state. In the default state, all sliders were at their top limit, all switches to the left, and all knobs rotated to the counter-clockwise limit. Participants were shown PowerPoint slides before each condition describing the tasks and instructions for the synchronization method. Participants were required to use only their dominant hand to complete the tasks.

When the participant was ready to begin a condition, they put on the HMD and were placed into the virtual environment shown in Fig. 10. To begin each group of tasks, the participant first pressed the center button aligned with the single physical button on the interface. Then the participant must return their hand to the origin to realign their physical and virtual hand and begin the study. Participants were also encouraged by the study supervisor to rest their hands on the surface after completing the reset while they selected the next target using the head pointer.

At the start of each group of three tasks, the corresponding layout for the control type as shown in Fig. 8. Participants were free to complete the interaction tasks in any order within each group. To determine the order, they used the head pointer to select the control and could then complete the task. For this study, the

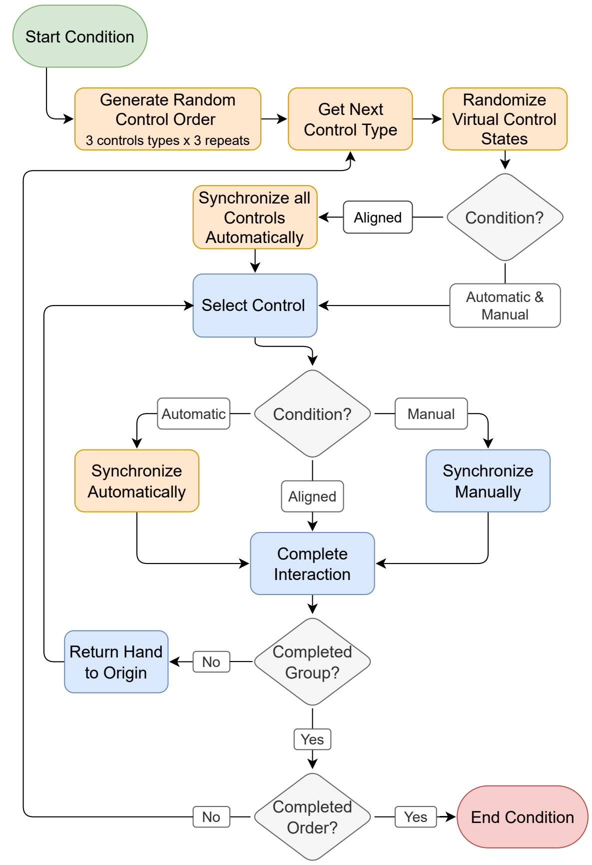


Figure 9: The flow for a task sequence in the study procedure. Blue boxes indicate processes performed by the user, while orange indicates the processes completed by the system.

selection area around the switches and knobs was set at 4 cm x 4 cm, while the selection area for the slider was a 3 cm x 12 cm rectangle.

Once a task has been selected, the manual or automatic state synchronization will activate. In every set of tasks, the virtual controls started started in randomized positions. As a result, in the Aligned condition, automatic synchronization was applied to ensure alignment between the random state of the virtual control and the physical control. In the Manual-Remapped condition, the participant first matched the physical state (visualized as an orange copy of the control, overlaid on the virtual control) to the state of the selected virtual control (visualized as the virtual control with blue semi-transparent color). In contrast, in the Auto-Remapped condition, there would be a brief moment where the state changes, during which the text "Awaiting State Sync" would appear.

Upon successful synchronization (automatic or manual), the control would turn white, and the user can complete the task. Successful completion of each task was indicated with a green rectangle that appeared around the task. When the task was completed and the user returned their hand to the origin area, the selection of the completed task was disabled, and they could continue by selecting the next task. Upon completing all tasks in a group, the user is shown

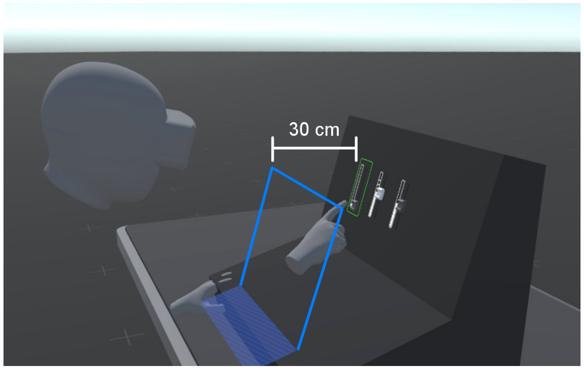


Figure 10: The virtual environment for the study. The blue rectangle indicates the origin area where users should put their hands between tasks in which the physical and virtual hands are aligned.

a single button and prompted to press it when they are ready to continue to the next group of tasks. The procedural flow of each condition is outlined in Fig. 9.

For training, participants completed the full task procedure twice for each control in the order Switch, Knob, Slider. This enabled the participant to practice the task and allowed them to adapt to the experience of haptic retargeting. After the training, the participant is asked if they are ready to continue.

Once the participant stated they were ready to continue, the task procedure was repeated in a random order eight times per control type for a total of 24 groups of three tasks per condition (excluding training tasks). Once all task groups were complete, the participants were asked to remove the VR headset and complete the post-condition questionnaire.

5 RESULTS

For the analysis of task completion time and incorrect interactions, linear mixed-effect models were conducted using the 'lme4' package in R. Type II Wald chi-square tests using the 'car::Anova' function were used to extract significance, and pairwise posthoc comparisons were conducted using Tukey's HSD for multiple comparisons where appropriate. This analysis was chosen as linear mixed effect models can account for variance between and within individual participants in repeated measures designs. Wald chi-square tests are suitable for extracting the significance of fixed effects [14]. Responses to the questionnaires were analyzed using Kruskal-Wallis one-way ANOVA. Pairwise comparisons were conducted using the Dunn test with p-values adjusted using the Holm-Bonferroni method.

While the center tasks in the Auto-Remapped and Manual-Remapped conditions are aligned with the center physical control, these data points are still analyzed with the left and right tasks that require retargeting. In the Aligned condition, the physical and virtual control states match. In contrast, in the other conditions with the remapped interface, the state of the center virtual control will be different to the center physical control, requiring the state conflict to be resolved either automatically or manually by the user.

5.1 Task Completion Time

The linear mixed-effect models were conducted separately for each control type for task completion time. The models were specified with the condition's fixed effect and the participant's random effect on the intercept. For all three controls types, the models showed a significant fixed effect of Condition (Slider: $\chi^2(2) = 411.45, p < 0.0001$,, Knob: $\chi^2(2) = 516.46, p < 0.0001$, Switch: $\chi^2(2) = 463.65, p < 0.0001$), see Fig. 11. For every control, pairwise comparisons showed task completion time was significantly lower in the Aligned condition than in the Auto-Remapped ($p < 0.0001$) and Manual-Remapped ($p < 0.0001$) conditions. Likewise, completion times for Auto-Remapped were significantly lower than Manual-Remapped ($p < 0.0001$).

5.2 Incorrect Interactions

Unlike task completion time, where the nature of the task is different for each control type, the reach to grasp action where errors are likely to occur is similar for each control type. As such, the analysis is conducted between control types. For incorrect interactions, the models were specified with the fixed effects of the Condition and the Control type with the interaction effect between them and the random effect of the participant on the intercept. The model showed a significant fixed effect of control type ($\chi^2(1) = 411.45, p < 0.0001$). Pairwise comparisons between Control types showed that the number of incorrect interactions was significantly lower for the Switch than the Knob ($p = 0.0392$) controls. No significant differences were found between the Slider and the Switch ($p = 0.7598$) or Knob ($p = 1.888$) (See Fig. 12).

5.3 Questionnaires

The raw NASA TLX responses were analyzed as separate questions with significant differences found for mental demand ($\chi^2(2) = 7.4294, p = 0.02436$), effort ($\chi^2(2) = 7.9182, p = 0.01908$) and Frustration Level ($\chi^2(2) = 5.853, p = 0.05358$) (See Fig. 13).

Pairwise comparisons showed mental demand was significantly higher for Manual-Remapped than for Aligned ($p = 0.02$). For effort, Manual-Remapped was significantly higher than Aligned ($p = 0.019$) with no other significant comparisons. No significance was found for the Frustration Level.

Analysis of the System Usability Scale scores showed significant differences between conditions ($\chi^2(2) = 13.424, p = 0.0012$) with pairwise comparisons showing a significant difference between the Aligned and Manual-Remapped conditions ($p = 0.00078$) and no significant difference between Auto-Remapped and Manual-Remapped or Aligned (See Fig. 13).

For the perception questions, significant differences were found for the questions on Ownership ($\chi^2(2) = 9.0834, p = 0.01066$), Position ($\chi^2(2) = 463.65, p < 0.0001$), Movement ($\chi^2(2) = 463.65, p < 0.0001$), and Alignment (($\chi^2(2) = 463.65, p < 0.0001$) (See Fig. 13). Pairwise comparisons for Ownership showed users felt more ownership of their hand in the Aligned condition than Auto-Remapped ($p = 0.018$) and Manual-Remapped ($p = 0.028$) conditions. Likewise, they felt their hand position and movement were more accurate in the Aligned condition than the Auto-Remapped (Position:

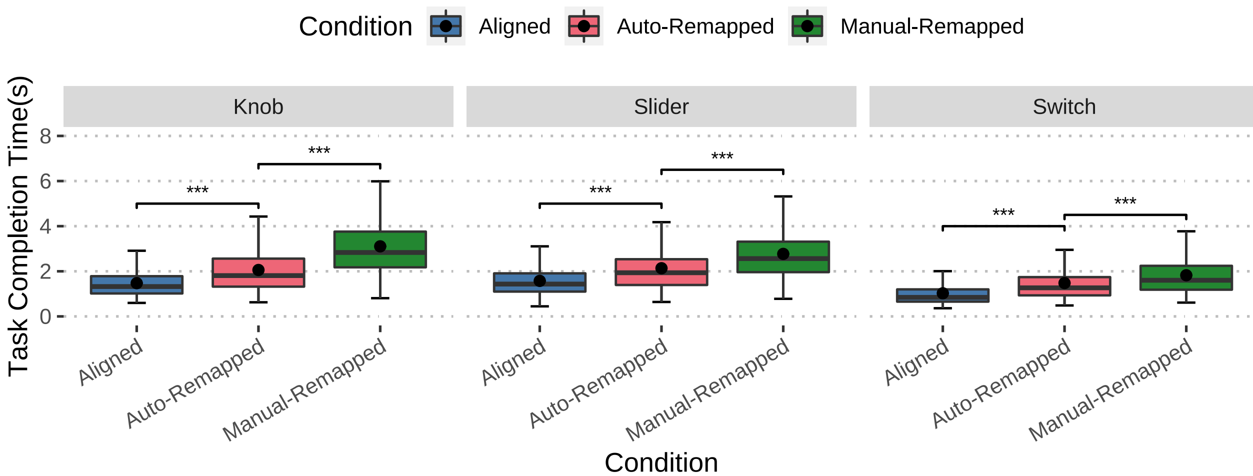


Figure 11: Task completion time for each task/control type. Only significant differences that were found between Aligned and Auto-Remapped, and Auto-Remapped and Manual-Remapped are shown.

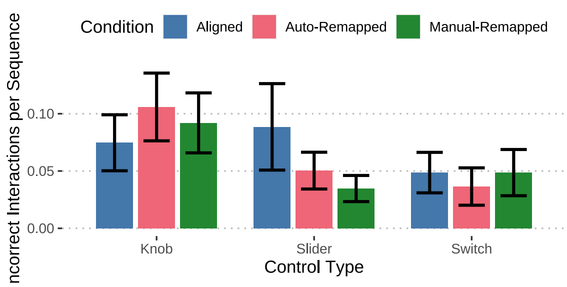


Figure 12: Incorrect interactions for control types in each condition, error bars indicate ± standard error. Significant differences were found between the combined incorrect interactions in Knob and Switch (*).

p = 0.0028, Movement:p = 0.0007) and Manual-Remapped (Position: p = 0.0060, Movement: p = 0.0007) conditions. Finally, participants felt the interface they saw most aligned with the interface they touched in the Aligned condition compared to the Remapped conditions (Auto:p = 0.0191, Manual:p = 0.0068).

5.4 Preferences and Feedback

Of the 24 participants, only four preferred the remapped interface over the aligned interface. Two of these four participants reported being unable to identify a difference, and one commented that they felt they were "...moving more for the aligned [interface]." Of the remaining 20 participants that preferred the aligned interface, the overall sentiment was that it felt more natural and it provided better interaction accuracy and speed.

Similarly, four participants preferred manual over automatic synchronization (all different participants from those that preferred Retargeting). One of the four participants who preferred manual synchronization commented that the sound of the automatic synchronization broke their immersion, and two commented that it felt more natural and realistic. Overall, the participants that preferred the automatic synchronization reported that it took "less work" and "less effort" while manual synchronization was "confusing" and "more cumbersome."

Interestingly one participant commented that they noticed the retargeting more with the manual synchronization. In particular, they observed that the physical control visualization was always in the same position as they left the previous control, highlighting that they were re-using the same control repeatedly and not separate controls.

5.5 Task Completion Order

Given the participants were free to complete the tasks within each group in any order, this may have introduced some noise into the data. Participants knew they were manually correcting the same component in the Manual-Remapped condition. So they could theoretically minimize the amount of correction needed by carefully planning the order they completed the tasks. To understand if there was an effect, post hoc data exploration and analysis were conducted.

Participants were generally observed to favour one particular order: Left, Center, Right (LCR). This is supported by the data, which showed that in all three conditions, 87.7% of tasks were completed using the LCR order. The 12.3% of tasks completed in alternative orders to LCR were spread across 12 of the 24 participants.

Given the inconsistent distribution of orders, it is not fair to directly analyze if there was an effect of the task order; instead, we look at the distribution of task orders between conditions to identify and investigate if participants acted in a similar or attempted to optimize their task order in the Manual-Remapped condition.

We computed the cosine similarity between the distributions of task orders for each pair of conditions using the 'lsa' package

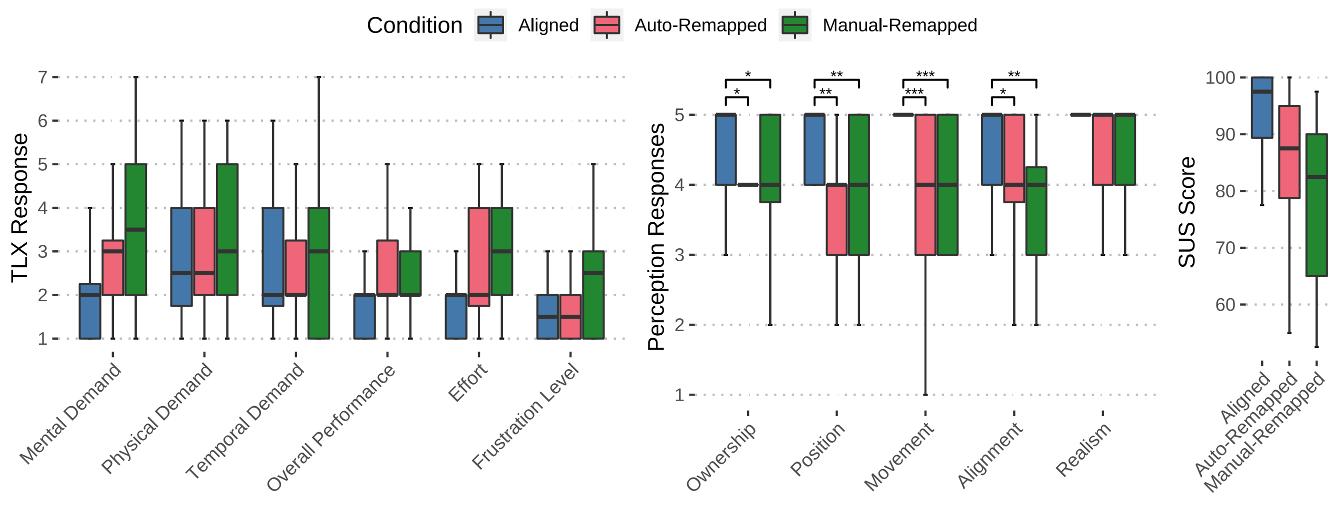


Figure 13: Questionnaire responses and calculated System Usability Scale scores. The only significant differences found were between Aligned and Auto-Remapped, and Auto-Remapped and Manual-Remapped are shown.

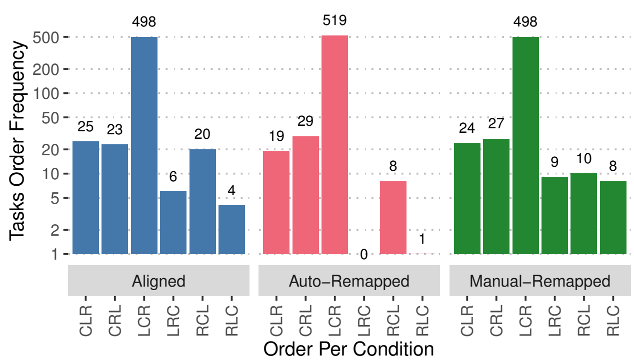


Figure 14: The distribution of the orders in which participants completed the tasks for each condition.

in R. Cosine similarity tests are sensitive to disproportionately large frequencies. Hence, the frequencies were log-transformed to mitigate the effect of the disparity between LCR and the other orders. The cosine similarities between conditions are shown in Fig. 14. In general, the distribution of task completion orders was very similar between conditions (Aligned - Auto-Remapped: 0.9958, Aligned - Manual-Remapped: 0.9353, and Auto-Remapped - Manual-Remapped: 0.9921). The similarity in distribution and the heavy tendency towards the LCR order suggests that participants generally did not try to optimize in the Manual-Remapped condition, and any noise introduced into the data by different completion orders would be minimal.

6 DISCUSSION

In the study, the aligned interface was predicted to perform better in all metrics, considering it presents an ideal configuration of a physical-virtual interface. In general, this was confirmed by the findings of the study. This was not surprising as the aligned interface provides an ideal operating performance that remapped implementations can strive to achieve. However, we highlight that the aligned interface does not provide the flexibility of remapped interface. Although it performed better overall, the aligned interface cannot be changed dynamically without changing the physical interface layout. The remapped physical-virtual interface used in the Auto-Remapped and Manual-Remapped conditions can be virtually adjusted in real time and provide accurate haptic feedback to the user.

In general, the automatic synchronization method was shown to perform better for task completion time than the manual synchronization. It also resulted in less task load and better usability, though not significantly less. It was also the most preferred synchronization method. As such, it can be concluded that an ideal remapped physical-virtual interface would incorporate electro-mechanical controls that can adapt to the virtual control state. However, such a dynamic passive-haptic interface is not always accessible or viable. In that situation, the manua synchronization method could be applied to support haptic retargeted interface interactions while providing a usable system.

In this study, the Knob and Slider control always needed synchronization, even for small differences. The Switch control, on the other hand, required synchronization only 50.7% of the time. This would be expected to increase for a switch with more possible positions, such as a 'double throw' switch with three positions or a rotary switch. Electro-mechanically controlled switches are also more difficult to source than other controls. As such, where they are unavailable, manua synchronization could support remapped switches with automatic synchronization for more accessible knobs and sliders.

6.1 Findings

H1 was confirmed in the study results, with task completion time being significantly lower without haptic retargeting. For the two synchronization methods, the extra step to re-align the controls predictably did result in significantly shorter task completion times for the automatic synchronization, confirming H2. H3 and H4 could not be confirmed, with the only significant difference in incorrect interactions between the Knob and the Switch and no significant difference with and without haptic retargeting. This is likely caused by the different types of interaction required and the sensitivities of the different physical components. It is much easier to bump or slightly move the Knob control than it is to switch the wrong switch accidentally.

H5 and H6 cannot be confirmed without significant differences in task load and usability between conditions. The median usability scores remained in the upper end of the SUS range (> 80), with the Aligned condition being the highest. This indicates that while remapped physical-virtual interfaces tend to be less useful than an aligned passive haptic interface, they are generally considered on the scale's usable end.

Regarding task load, the median mental demand tended to be lower without haptic retargeting (Aligned) than with (Auto-Remapped) but no significant differences as such H7 and H8 cannot be confirmed. No significant differences were found between the two synchronization methods, and as such, H9 cannot be confirmed. H10 was confirmed with the perceived accuracy of the participant's hands, and the interface was reduced by haptic retargeting.

These findings highlight that while remapped physical-virtual interfaces are considered usable, more work can be done to increase usability and reduce the effect of haptic retargeting on task completion time. There has been little prior work exploring the change in task completion times with and without haptic retargeting. Cheng et al. explored the effect of changing amounts of retargeting and found no significant effect [9], though the added complexity of the interface interactions may have exasperated this.

6.2 Limitations and Future Work

The procedural tasks used for the study are not necessarily representative of realistic interactions with virtual or physical control panels or interfaces. In future work, evaluations of remapped physical-virtual interfaces are required to quantify their performance using more realistic tasks. In the study, only stateful controls were explored, while the system could also support non-stateful controls like continuous knobs and buttons. In future work, we would like to explore interfaces that incorporate a variety of controls to achieve a more complex task.

Furthermore, some of the subjective results for the experiment lacked significant outcomes, though trends could be observed in the responses. More participants could have resulted in stronger statistical findings for task load and usability. A future investigation in which users are required to complete more complex tasks may also provide more meaningful insight into the effects of the synchronization method and haptic retargeting on interface usability.

Additionally, it is unclear if hand redirection and haptic retargeting impart muscle memory in the same manner that training operating a conventional physical interface does. We believe this is an interesting area of investigation as the movement to operate the controls is no longer fixed, so may be difficult to learn. A potential positive aspect is this could be used to avoid repetitive operations avoiding strain or overuse of specific muscle groups.

It may also be possible to reduce the amount of retargeting required by combining these controls with encounter type haptics [16], along with other retargeting methods like blink suppressed hand redirection [44]. Furthermore, linear scaling illusions can support a variety of virtual slider lengths [12]. It may also be possible to scale the rotation of knobs to create knobs with a variable range of rotation.

7 PROPOSED APPLICATIONS

While more work is required to increase usability, we outline some conceptual applications in which we envisage remapped physical-virtual interfaces could be applied. The extension of remapped physical-virtual interfaces to support stateful controls supports rich and diverse user interfaces, expanding the opportunities to apply the technology in new application areas. The manual synchronization method also allows the remapping of more ubiquitous interfaces with stateful controls that lack automatic actuation. For example, light switches and dimmers, audio mixers, and video game controllers could be opportunistically used for remapped interfaces.

7.1 Tangible VR Interfaces and Menu Systems

Contextual interfaces and menu systems are common elements of VR applications. Remapped physical-virtual interfaces with stateful controls could be incorporated into applications that provide multiple tools or functions, each with different settings. Drawing applications, CAD modeling, and sculpting applications could use tangible sliders to adjust various virtual parameters of the tools in use. Similarly, immersive analytics and data visualization systems could use tangible knobs for filtering data and adjusting scales.

7.2 Immersive Simulators and Training

Physical vehicle simulators are a critical training element for operators in the aviation, defense, and space industries. These simulators are usually expensive and specially tailored to the simulated vehicle model and variant. Using remapped physical-virtual interfaces, we propose that a generic physical simulator system can be extended to provide an immersive simulation of many vehicles of a similar type. For example, a single flight simulator with actuated switches and knobs could be remapped to the controls of multiple airplane cockpits. This would enable more flexibility in the training possible with a single simulator reducing the need for multiple specially tailored simulators.

7.3 User Configurable Interfaces and Ergonomics

Remapped interfaces allow users to configure a virtual interface layout to suit their preferences and ergonomic requirements. Some control panels, such as factory control systems and studio audio mixing consoles, are large and cumbersome. Physical controls can be within a comfortable reach volume, enabling users to operate a much larger virtual interface and meet their ergonomic requirements. Previous work has used hand redirection to adjust physical reach distances and improve ergonomics in virtual environments [32]. These techniques could be adapted into the remapped physical-virtual interface system to retain the spatial layout of the full-sized interface and the haptic feedback while reducing the space requirements and the necessary reaching distances.

Alternatively, given the limited range of detectable and tolerable hand redirection, multiple separate virtual interfaces around a room could also be docked to a single fixed or handheld physical interface [31]. A user could either bring the physical interface to the virtual interface and link them together or vice-versa.

7.4 Impossible Interfaces and Portals

Remapping physical interface controls to dynamic virtual reality interfaces allows a tangible interface to escape the confines of what is possible in the physical world. Haptic retargeting could be combined with portals to support interaction with tangible virtual interfaces at a distance or between rooms. Portals can bring out-of-reach objects into the physical reach volume of the user [36]. A natural next step is to introduce haptic retargeting and enable tangible interactions with the out-of-reach virtual interface. Another impossible interface could involve distorting the interface itself. The virtual interface could be distorted with a lens-like effect that expands the immediate space near the hand. This would allow fine-grain interactions with tangible controls while providing a nearby overview of a much larger interface.

8 CONCLUSION

This paper presents techniques to support the integration of haptic retargeting with stateful physical and virtual controls on remapped physical-virtual interfaces. Two methods for handling state conflicts were proposed and explored through a user study: an automatic method using electro-mechanical controls that a computer could adjust and a manual method that relied on user correction. In addition, the remapped physical-virtual interface approach was compared to a virtual interface with aligned passive haptics. While the aligned interface is the ideal implementation of a passive haptic interface, participants still found the remapped physical-virtual interface usable. automatic synchronization provided better task performance and was preferred over manual synchronization; however, the automatic method relies on more complex dynamic passive-haptic controls, which may be inaccessible in some applications. Enabling realistic haptics in VR is complex and remains an open research challenge. The techniques and system presented in this paper help build toward the creation of rich, flexible, and usable remapped physical-virtual interfaces with real haptic feedback and affordances for VR experiences.

ACKNOWLEDGMENTS

This research is funded by an Australian Research Training Program Scholarship (RTP).

REFERENCES

[1] Parastoo Abtahi and Sean Follmer. 2018. Visuo-Haptic Illusions for Improving the Perceived Performance of Shape Displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. ACM Press, 1–13. https://doi.org/10.1145/3173574.3173724

[2] Bruno Araujo, Ricardo Jota, Varun Perumal, Jia Xian Yao, Karan Singh, and Daniel Wigdor. 2002. Snake Charmer: Physically Enabling Virtual Objects. *Architectural Review* 1267 (2002), 85–86. https://doi.org/10.1145/2839462.2839484

[3] Ferran Argelaguet and Carlos Andujar. 2009. Efficient 3D Pointing Selection in Cluttered Virtual Environments. *IEEE Computer Graphics and Applications* 29, 6 (Nov. 2009), 34–43. https://doi.org/10.1109/MCG.2009.117 Conference Name: IEEE Computer Graphics and Applications.

[4] Ferran Argelaguet and Carlos Andujar. 2013. A survey of 3D object selection techniques for virtual environments. *Computers & Graphics* 37, 3 (May 2013), 121–136. https://doi.org/10.1016/j.cag.2012.12.003

[5] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2016. Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*. ACM Press, 1968–1979. https://doi.org/10.1145/2858036.2858226

[6] Yuki Ban, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. 2014. Displaying shapes with various types of surfaces using visuo-haptic interaction. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology (VRST '14)*. Association for Computing Machinery, New York, NY, USA, 191–196. https://doi.org/10.1145/2671015.2671028

[7] John Brooke. 1995. SUS: A quick and dirty usability scale. *Usability Eval. Ind.* 189 (Nov. 1995).

[8] Li-Wei Chan, Hui-Shan Kao, Mike Y. Chen, Ming-Sui Lee, Jane Hsu, and Yi-Ping Hung. 2010. Touching the void: direct-touch interaction for intangible displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. Association for Computing Machinery, New York, NY, USA, 2625–2634. https://doi.org/10.1145/1753326.1753725

[9] Lung-Pan Cheng, Eyal Ofek, Christian Holz, Hrvoje Benko, and Andrew D. Wilson. 2017. Sparse Haptic Proxy: Touch Feedback in Virtual Environments Using a General Passive Prop. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press, 3718–3728. https://doi.org/10.1145/3025453.3025753

[10] Aldrich Clarence, Jarrod Knibbe, Maxime Cordeil, and Michael Wybrow. 2021. Unscripted Retargeting: Reach Prediction for Haptic Retargeting in Virtual Reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. 150–159. https://doi.org/10.1109/VR50410.2021.00036 ISSN: 2642-5254.

[11] Nathan Cournia, John D. Smith, and Andrew T. Duchowski. 2003. Gaze- vs. hand-based pointing in virtual environments. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems (CHI EA '03)*. Association for Computing Machinery, New York, NY, USA, 772–773. https://doi.org/10.1145/765891.765982

[12] Martin Feick, Niko Kleer, André Zenner, Anthony Tang, and Antonio Krüger. 2021. Visuo-haptic Illusions for Linear Translation and Stretching using Physical Proxies in Virtual Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3411764.3445456

[13] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: Dynamic Physical Affordances and Constraints Through Shape and Object Actuation. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. ACM, 417–426. http://doi.acm.org/10.1145/2501988.2502032

[14] John Fox. 2015. *Applied regression analysis and generalized linear models*. Sage Publications.

[15] J. J. Gibson. 1933. Adaptation, after-effects, and contrast in the perception of curved lines. *Journal of Experimental Psycholology* 16, 1 (1933), 1–31. https://doi.org/10.1037/h0074626

[16] Eric J. Gonzalez, Parastoo Abtahi, and Sean Follmer. 2020. REACH+: Extending the Reachability of Encountered-type Haptics Devices through Dynamic Redirection in VR. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 236–248. https://doi.org/10.1145/3379337.3415870

[17] Dustin T. Han, Mohamed Suhail, and Eric D. Ragan. 2018. Evaluating remapped physical reach for hand interactions with passive haptics in virtual reality. *IEEE Transactions on Visualization and Computer Graphics* 24, 4 (April 2018), 1467–1476. https://doi.org/10.1109/TVCG.2018.2794659

[18] Chris Harrison and Scott E. Hudson. 2009. Providing dynamically changeable physical buttons on a visual display. In *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*. ACM Press, 299–299. https://doi.org/10.1145/1518701.1518749

[19] Sandra G. Hart. 2006. Nasa-Task Load Index (NASA-TLX); 20 Years Later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 50, 9 (Oct. 2006), 904–908. https://doi.org/10.1177/154193120605000909 Publisher: SAGE Publications Inc.

[20] Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2012. Tangible remote controllers for wall-size displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2865–2874. https://doi.org/10.1145/2207676.

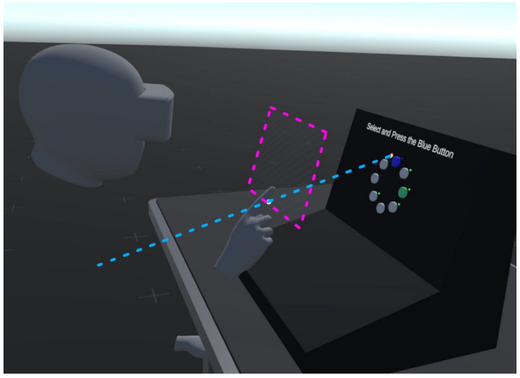
2208691
[21] Ricardo Jota, Miguel A. Nacenta, Joaquim A. Jorge, Sheelagh Carpendale, and Saul Greenberg. 2010. A comparison of ray pointing techniques for very large displays. In *Proceedings of Graphics Interface 2010 (GI '10)*. Canadian Information Processing Society, CAN, 269–276.
[22] Luv Kohli, Mary C. Whitton, and Frederick P. Brooks. 2012. Redirected touching: The effect of warping space on task performance. In *IEEE Symposium on 3D User Interfaces 2012, 3DUI 2012 - Proceedings*. 105–112. https://doi.org/10.1109/3DUI.2012.6184193
[23] Luv Kohli, Mary C. Whitton, and Frederick P. Brooks. 2013. Redirected Touching: Training and adaptation in warped virtual spaces. In *IEEE Symposium on 3D User Interface 2013, 3DUI 2013 - Proceedings*. IEEE, 79–86. https://doi.org/10.1109/3DUI.2013.6550201
[24] Byungjoo Lee and Antti Oulasvirta. 2016. Modelling Error Rates in Temporal Pointing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, San Jose, California, USA, 1857–1868. https://doi.org/10.1145/2858036.2858143
[25] R.W. Lindeman, J.L. Sibert, and J.K. Hahn. 1999. Hand-held windows: towards effective 2D interaction in immersive virtual environments. In *Proceedings IEEE Virtual Reality (Cat. No. 99CB36316)*. 205–212. https://doi.org/10.1109/VR.1999.756952 ISSN: 1087-8270.
[26] R.W. Lindeman, J.L. Sibert, and J.N. Templeman. 2001. The effect of 3D widget representation and simulated surface constraints on interaction in virtual environments. In *Proceedings IEEE Virtual Reality 2001*. 141–148. https://doi.org/10.1109/VR.2001.913780
[27] Robert W. Lindeman, John L. Sibert, and James K. Hahn. 1999. Towards Usable VR: An Empirical Study of User Interfaces for lmmersive Virtual Environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99*. ACM Press, 64–71. https://doi.org/10.1145/302979.302995
[28] Brandon Matthews, Bruce H Thomas, Stewart Von Itzstein, and Ross Smith. 2021. Adaptive Reset Techniques for Haptic Retargeted Interaction. *IEEE Transactions on Visualization and Computer Graphics* (2021), 1–1. https://doi.org/10.1109/TVCG.2021.3120410 Conference Name: IEEE Transactions on Visualization and Computer Graphics.
[29] Brandon J. Matthews and Ross T. Smith. 2019. Head Gaze Target Selection for Redirected Interaction. In *SIGGRAPH Asia 2019 XR (SA '19)*. Association for Computing Machinery, Brisbane, QLD, Australia, 13–14. https://doi.org/10.1145/3355355.3361883 event-place: Brisbane, QLD, Australia.
[30] Brandon J. Matthews, Bruce H. Thomas, G. Stewart Von Itzstein, and Ross T. Smith. 2022. Shape Aware Haptic Retargeting for Accurate Hand Interactions. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 625–634. https://doi.org/10.1109/VR51125.2022.00083 ISSN: 2642-5254.
[31] Brandon J. Matthews, Bruce H. Thomas, Stewart Von Itzstein, and Ross T. Smith. 2019. Remapped Physical-Virtual Interfaces with Bimanual Haptic Retargeting. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 19–27. https://doi.org/10.1109/VR.2019.8797974 ISSN: 2642-5254.
[32] Roberto A. Montano Murillo, Sriram Subramanian, and Diego Martinez Plasencia. 2017. Erg-O: Ergonomic Optimization of Immersive Virtual Environments. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. Association for Computing Machinery, New York, NY, USA, 759–771. https://doi.org/10.1145/3126594.3126605
[33] Paul Myers, Arnold Starr, and Kadie Mullins. 2018. Flight Simulator Fidelity, Training Transfer, and the Role of Instructors in Optimizing Learning. *International Journal of Aviation, Aeronautics, and Aerospace* 5, 1 (Feb. 2018). https://doi.org/10.15394/ijaaa.2018.1203
[34] Donald A Norman. 1988. *The Psychology of Everyday Things*. Basic Booksss.
[35] Matthias Oberhauser, Daniel Dreyer, Reinhard Braunstingl, and Ioana Koglbauer. 2018. What's Real About Virtual Reality Flight Simulation? *Aviation Psychology and Applied Human Factors* 8, 1 (March 2018), 22–34. https://doi.org/10.1027/2192-0923/a000134 Publisher: Hogrefe Publishing.
[36] Henning Pohl, Klemen Lilija, Jess McIntosh, and Kasper Hornbæk. 2021. Poros: Configurable Proxies for Distant Interactions in VR. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3411764.3445685
[37] Shane R Porter, Michael R Marner, Ross T Smith, Joanne E Zucco, and Bruce H Thomas. 2010. Validating spatial augmented reality for interactive rapid prototyping. In *Mixed and augmented reality (ISMAR), 2010 9th IEEE international symposium on*. IEEE, 265–266.
[38] Helena Roeber, John Bacus, and Carlo Tomasi. 2003. Typing in thin air: the canesta projection keyboard - a new method of interaction with electronic devices. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems (CHI EA '03)*. Association for Computing Machinery, Ft. Lauderdale, Florida, USA, 712–713. https://doi.org/10.1145/765891.765944
[39] Daniel Schneider, Verena Biener, Alexander Otte, Travis Gesslein, Philipp Gagel, Cuauhtli Campos, Klen Čopič Pucihar, Matjazz Kljun, Eyal Ofek, Michel Pahud, Per Ola Kristensson, and Jens Grubert. 2021. Accuracy Evaluation of Touch Tasks in Commodity Virtual and Augmented Reality Head-Mounted Displays. In *Symposium on Spatial User Interaction (SUI '21)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3485279.3485283
[40] Tim M. Simon, Ross T. Smith, Bruce Thomas, Stewart Von Itzstein, Mark Smith, Joonsuk Park, and Jun Park. 2012. Merging tangible buttons and spatial augmented reality to support ubiquitous prototype designs. In *Proceedings of the Thirteenth Australasian User Interface Conference - Volume 126 (AUIC '12)*. Australian Computer Society, Inc., Melbourne, Australia, 29–38.
[41] Markus Tatzgern and Christoph Birgmann. 2021. Exploring Input Approximations for Control Panels in Virtual Reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. 1–9. https://doi.org/10.1109/VR50410.2021.00092 ISSN: 2642-5254.
[42] Malte Weiss, Florian Schwarz, Simon Jakubowski, and Jan Borchers. 2010. Madgets: actuating widgets on interactive tabletops. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10)*. Association for Computing Machinery, New York, New York, USA, 293–302. https://doi.org/10.1145/1866029.1866075
[43] Eric Whitmire, Hrvoje Benko, Christian Holz, Eyal Ofek, and Mike Sinclair. 2018. Haptic Revolver: Touch, Shear, Texture, and Shape Rendering on a Reconfigurable Virtual Reality Controller. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18* (2018), 1–12. https://doi.org/10.1145/3173574.3173660
[44] Andre Zenner, Kora Persephone Heqitz, and Antonio Kruger. 2021. Blink-Suppressed Hand Redirection. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. IEEE, Lisboa, Portugal, 75–84. https://doi.org/10.1109/VR50410.2021.00028
[45] André Zenner and Antonio Krüger. 2019. Estimating Detection Thresholds for Desktop-Scale Hand Redirection in Virtual Reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 47–55. https://doi.org/10.1109/VR.2019.8798143 ISSN: 2642-5254.
[46] André Zenner, Kristin Ullmann, and Antonio Krüger. 2021. Combining Dynamic Passive Haptics and Haptic Retargeting for Enhanced Haptic Feedback in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (May 2021), 2627–2637. https://doi.org/10.1109/TVCG.2021.3067777 Conference Name: IEEE Transactions on Visualization and Computer Graphics.
[47] Yiwei Zhao and Sean Follmer. 2018. A Functional Optimization Based Approach for Continuous 3D Retargeted Touch of Arbitrary, Complex Boundaries in Haptic Virtual Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. ACM Press, 1–12. https://doi.org/10.1145/3173574.3174118

A EVALUATION OF POINTER BASE SELECTION METHODS FOR HAPTIC RETARGETING

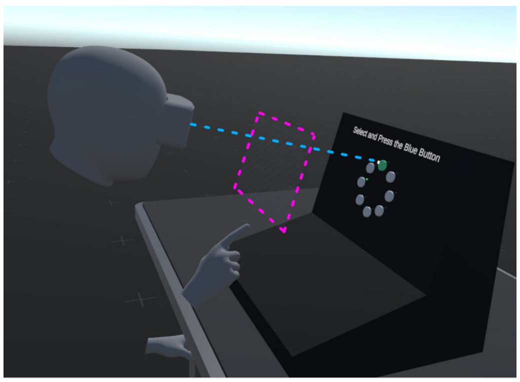
A.1 Target Selection Implementation

The *head pointer* method is a ray extending from the center of the user's view as measured from the HMD position in the direction it is facing. The *hand pointer* is based on the pointer implemented in the Oculus Quest hand tracking system. This implementation was selected based on initial testing that indicated it was more stable and easier to control than a pointer that extended from the fingertip. Fig. 15b shows the implementation within the virtual environment, annotated to show the head pointer ray.

For the hand pointer method, a ray extends from an approximated shoulder position, through a point between the user's thumb and index finger, with a reticle drawn at the point the ray intersects with the interface. The position of the user's shoulder is estimated as 20cm down from the eye position and 20cm perpendicular from the down and facing directions of the head. A reticle is drawn when the selection ray intersects with the interface or the controls. For this study, a dwell time of 0.3s was added to reduce the number of incorrect selections as the user moves the pointers. Fig. 15b shows the hand pointer implementation, annotated to show the hand pointer ray originating from the approximated shoulder position.



(a) Hand Pointer Method



(b) Head Pointer Method

Figure 15: The implemented pointer-based target selection methods explored in the preliminary target selection study and the virtual study environment. The blue line indicates the selection ray, and the pink rectangle outlines the semi-transparent plane that visualizes the rectangle that marked the origin.

A.2 Design and Participants

This preliminary study aims to evaluate pointer-based selection methods in the context of haptic retargeting and identify a method appropriate for the main study. The predictive methods have a non-zero chance of predicting the incorrect target, and thus they have been excluded from this study as they would not be suitable for use in this study. As such, this study compares only the *head pointer* and *hand pointer* selection methods seeking the more appropriate option for the main study.

For this study, 12 participants were recruited, ten male and two female, aged 24 to 36 (mean:27.9, σ:3.49). All participants self-reported as being predominantly right-handed with normal or corrected to normal vision. Participants were shown the selection methods using a slideshow and completed one full sequence of tasks to practice using the selection methods. A within-participants design was used with the two conditions fully counterbalanced, with half of the participants starting with the Head Pointer and the other half starting with the Hand Pointer. Of the participants, four reported using VR systems once or more per day, four weekly, two use them monthly, and two use them yearly.

Table 2: Button layout configurations with resulting proximity and retargeting gain for the preliminary study

Layout	*Diameter*	*Proximity*	*Angle Gain*	*Translation Gain*
Close	8 cm	3.06 cm	7.595°	1.009
Medium	12 cm	4.59 cm	11.31°	1.019
Far	16 cm	6.12 cm	14.931°	1.035

The apparatus, physical environment, virtual environment, and calibration procedure used in this study were identical to those used in the main study (Section. 4.4). The physical interface consisted of only a single central tactile button. The virtual interface was a ring of 8 buttons centered on the physical button. Three layouts were tested: Close, Medium, and Far. Each layout was a ring centered around the physical button with different diameters: 8 cm, 12 cm, and 16 cm respectively. These layouts produce different retargeting gain and target proximity to quantify the effect on the selection methods. The parameters of the different conditions are outlined in Table 2.

A.3 Measures and Hypotheses

The key measures in this study were selection and interaction time, incorrect selections, task load, system usability, and preferences. An effective target selection technique would ideally have minimal errors, minimal selection time, and not significantly increase the interaction time. It should also be considered usable and not overly tiring for the user.

Selection time was measured as the time between the return of the participant's hand into the threshold origin area and the correct target selection. Interaction time was measured as the time between the hand leaving the origin and pressing the selected button. Incorrect selections were summarized as the number of selections in each task sequence in which the participant selected at least one incorrect target before selecting the correct target. It was hypothesized that the head pointer will result in a lower selection time while the hand pointer would result in a lower interaction time.

Task load was collected using the raw NASA TLX questionnaire [19] on a 7-point Likert scale, and system usability was measured with the System Usability Scale [7]. It was hypothesized that physical demand would be higher for the hand pointer, but the hand pointer will be more usable. Participants were also asked to rate their speed and accuracy on a 5-point Likert scale (1:Slow to 5:Fast and 1:Low to 5:High, respectively).

A.4 Procedure

Participants were provided with a PowerPoint presentation describing haptic retargeting, the task, and the target selection methods used. Participants were then placed into the virtual world, and the interface was calibrated. Participants then completed one full sequence of tasks with each layout as the training procedure.

Participants first pressed the button aligned with the single button on the physical interface in each task sequence to begin the sequence. The interface was then changed to show the virtual task layout. Participants were required to select and press each button in the ring in a pre-determined, randomized order. The target button was highlighted in blue, and upon a successful selection, it would turn green. If the participant selected the wrong button, that button would highlight red, and interaction was prevented interactions until the selection of the correct button. Once the correct button had been selected, the participant pressed it and returned their hand to the target selection area behind the threshold. Between individual tasks in a sequence, participants were required to complete a static threshold type reset [28] where the origin area is visualized by a semi-transparent grey rectangle 30cm from the virtual interface. Each task sequence was repeated randomly for the three layouts (Close, 12 cm, 16 cm).

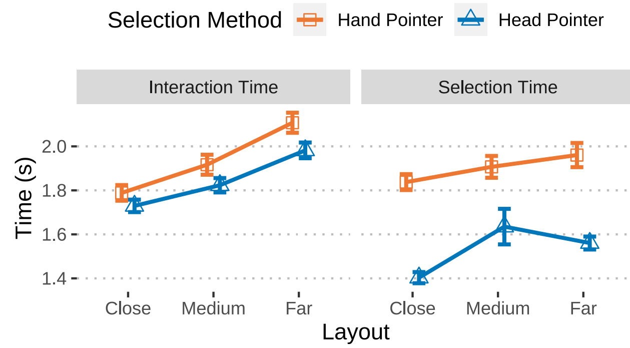
A.5 Results and Discussion

Linear mixed-effects models were conducted to analyze selection time, interaction time, and incorrect selections using the lme4 package in R. The models were specified with the fixed effects of the Selection Method and Layout, including an interaction effect between them and a random effect of the participant on the intercept. Type II Wald chi-square tests were used to extract significance, and pairwise posthoc comparisons were conducted using Tukey's HSD for multiple comparisons where appropriate. Responses to the questionnaires were analyzed using Mann-Whitney U tests.

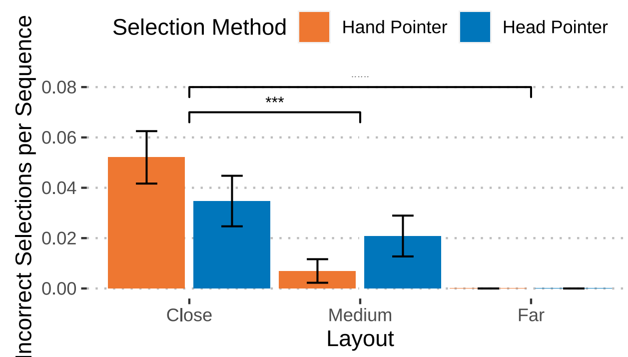
A.5.1 Selection and Interaction Time. For selection time shown in Fig. 16a, the model showed a significant fixed effect of the Selection Method ($\chi^2(1) = 128.9896, p < 0.0001$) with selection times being significantly lower for head pointer than for hand pointer. A significant fixed effect of Layout ($\chi^2(2) = 12.6053, p = 0.001831$) was also shown, and no significant interaction effect was found. Pairwise comparisons on the Layout indicated the Close Layout resulted in a significantly lower selection time than 12 cm ($t = -2.866, p = 0.0117$) and 16 cm ($p = 0.0033$). Selection times for Medium and Far were not significantly different ($p = 0.9099$).

For interaction time shown in Fig. 16a, the model showed a significant fixed effect of the Selection Method ($\chi^2(1) = 16.0715, p < 0.0001$) with the selection head pointer significantly faster hand pointer. A significant fixed effect of Layout ($\chi^2(2) = 126.0190, p < 0.0001$) was also shown, and no significant interaction effect was found. Pairwise comparisons on the Layout indicated that the Close Layout resulted in significantly faster interaction time than 12 cm ($p = 0.0001$) and 16 cm ($p < 0.0001$). Interaction times for Medium were also significantly lower than Far ($p < 0.0001$).

These results contradict our initial hypothesis that the hand pointer would be faster due to the hand being retracted to the origin area when completing the reset. This improved performance of the head pointer approach could be explained partly because the user could see the next target as soon as they finished the interaction with their current target, meaning they could begin turning toward the next target while returning their hand to the origin area. Furthermore, for the layout of controls, selection and interaction time were significantly lower for the Close layout, which is to be expected with the shorter travel time.



(a) Selection Time



(b) Incorrect Selections

Figure 16: Interaction time and selection time (a) and incorrect selections (b). For incorrect selections, error bars indicate ± standard error

A.5.2 Incorrect Selections. The model for incorrect selections showed a significant fixed effect of Layout ($\chi^2(1) = 16.0715, p < 0.0001$), with no other significant effects. Pairwise comparisons showed the rate of errors for the Close Layout was significantly higher than the 12 cm ($p = 0.0002$) and 16 cm ($p < 0.0001$). The 12 cm and 16 cm Layout error rates were not significantly different ($p = 0.1268$). (See Fig. 16b).

Despite the Close layout being faster, it did result in more errors. This highlights a challenge for target selection and haptic retargeting, particularly when the goal is for the illusion to remain undetectable. The detectable range is quite small, and therefore reused controls need to be very close together, which, in this study, was shown to result in more selection errors.

The accuracy of both techniques was more than 99.94% and 100% for both techniques with the 16 cm diameter layout and a spacing of 6.12 cm. This accuracy is much higher than the predictive techniques explored in previous literature [9, 10], but it requires extra effort and input from the user. Future work could explore pointer-based techniques compared to predictive techniques in the same context to better evaluate their comparative accuracy.

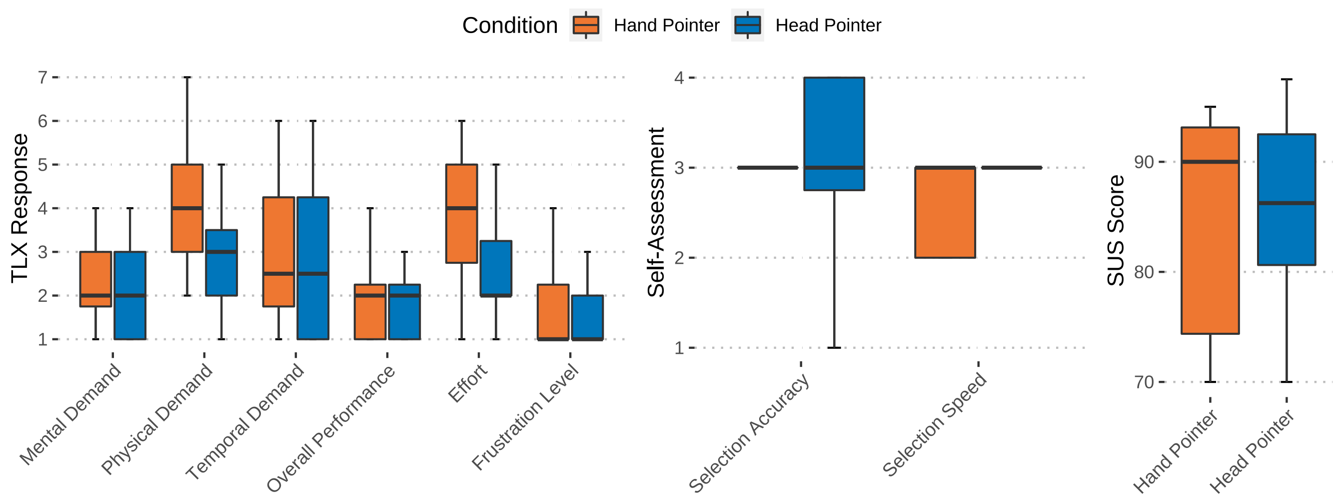


Figure 17: Questionnaire responses and calculated System Usability Scale scores for the preliminary target selection study.

A.5.3 Questionnaires. No significant differences were found in target selection speed and accuracy self-evaluation. The raw NASA TLX questionnaire was analyzed as separate questions. Though the hand pointer's physical demand and effort appeared high, no significant differences were found. Finally, the System Usability Scale scores also showed no significant difference. Fig. 17 shows the response data for the subjective questionnaires. Eight participants preferred the head pointer method, while the remaining four preferred the hand pointer.

Furthermore, while no significant differences were found in the TLX categories or the usability, it was observed that the median physical demand and effort were higher for the hand pointer than the head pointer. However, the hypothesis could not be confirmed. Of the eight participants that preferred the head pointer, six reported it being less tiring or fatiguing than the hand pointer. This was expected due to the requirement to hold their hand steady while selecting the target. While the threshold reset was the same for the head pointer, participants were observed not to move their hands as much while selecting the next target, and some rested their hands in their lap or on the desk between interactions. The increased fatigue with the hand pointer could be reduced by introducing some filtering to smooth out slight jitters in the movement, so the user is not required to hold their hand as steady [4].

Although not significantly different, the median usability score was higher for the hand pointer (90.0) than the head pointer (86.25), supporting but not confirming our hypothesis, and both scored above the "Good" level. More participants could have provided stronger statistical strength to the subjective feedback.