

Dynamic Redirection for VR Haptics with a Handheld Stick

Yuqi Zhou*
Purdue University

Voicu Popescu†
Purdue University

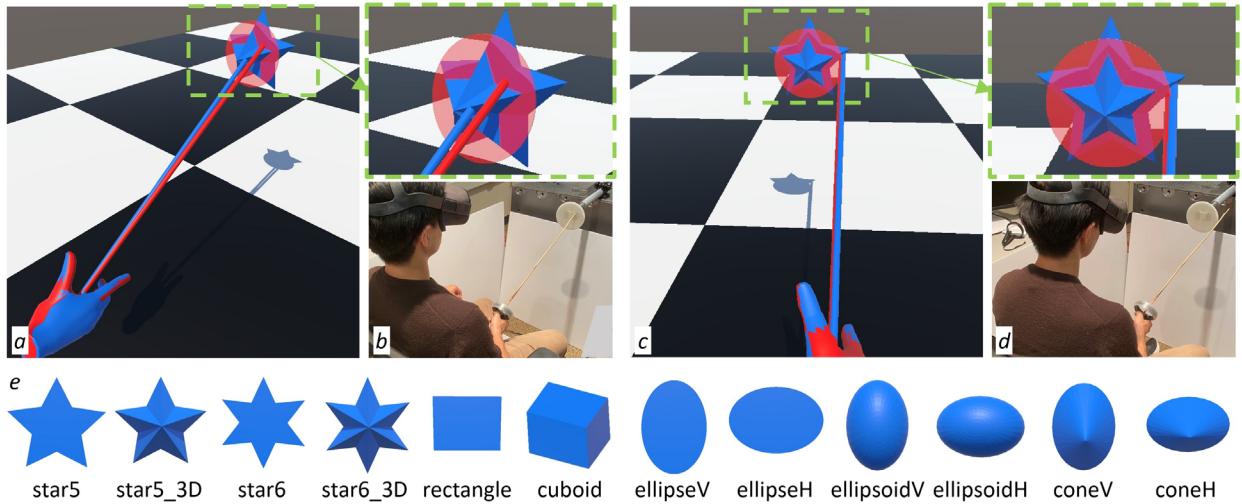


Fig. 1. Dynamic redirection of handheld stick to provide passive haptic feedback during 3D shape tapping (a) and contour tracing (c). Virtual entities (i.e., hand, stick, and object) are shown in blue, real entities are shown in red. The real entities are only visualized here for illustration purposes. The virtual hand and stick are redirected for the tip of the virtual stick to make contact with the inside (a) and contour (c) of the virtual 3D star-shaped object, as the real stick makes contact with the real object, i.e., a disk (b and d). Our haptic redirection method is general—it succeeds at providing convincing haptic feedback for a variety of virtual objects using the same real disk (e).

Abstract—This paper proposes a general handheld stick haptic redirection method that allows the user to experience complex shapes with haptic feedback through both tapping and extended contact, such as in contour tracing. As the user extends the stick to make contact with a virtual object, the contact point with the virtual object and the targeted contact point with the physical object are continually updated, and the virtual stick is redirected to synchronize the virtual and real contacts. Redirection is applied either just to the virtual stick, or to both the virtual stick and hand. A user study ($N = 26$) confirms the effectiveness of the proposed redirection method. A first experiment following a two-interval forced-choice design reveals that the offset detection thresholds are $[-15\text{cm}, +15\text{cm}]$. A second experiment asks participants to guess the shape of an invisible virtual object by tapping it and by tracing its contour with the handheld stick, using a real world disk as a source of passive haptic feedback. The experiment reveals that using our haptic redirection method participants can identify the invisible object with 78% accuracy.

Index Terms—Passive haptics, redirection, detection thresholds, virtual reality.

1 INTRODUCTION

In recent years, consumer-level virtual reality (VR) headsets have made great progress, allowing users of VR applications to see complex virtual worlds in vivid detail. However, the perception of the virtual world continues to rely mainly on the user's vision. Providing users with haptic feedback as they touch virtual objects remains challenging.

One approach for providing haptic feedback is to rely on real world objects aligned with the virtual objects with which the user makes contact. However, as real and virtual worlds are different, the opportunities for providing such passive haptic feedback are scarce. The approach of increasing the availability of passive haptic feedback by altering the

real world to match the virtual world is challenging—one would have to create physical replicas of all virtual worlds of interest of applications. On the other hand, the virtual world is more malleable—it is indeed easier to modify the virtual world to match the real world, through a process called redirection.

In haptic redirection, a real world object can serve as a substitute for a virtual object with a different shape. However, providing convincing haptic feedback to users as they touch a virtual object with their hands remains challenging. When we touch a real world object we perceive not just a simple reactionary force, but many other object properties such as intricate geometric detail, texture, and temperature. Therefore passive haptic redirection is hard pressed to fool the user's hands, even if the contact with the virtual and real objects are perfectly synchronized. If, on the other hand, the user explores the virtual world with a handheld stick, providing convincing haptic feedback becomes more tractable. When using a stick, users do not expect detailed haptic feedback, e.g., they do not expect the stick to convey texture or temperature. Consequently, if the haptic redirection algorithm succeeds at aligning the virtual object with a real object, the user will find plausible the haptic feedback provided by the collision between the stick and the real object. In other words, compared to the user's hand, the stick reduces the sensory acuity of the user and therefore reduces the discrepancy

*e-mail: zhou1168@purdue.edu

†e-mail: popescu@purdue.edu

between what the user expects and what the user feels.

Another benefit of using a stick to provide passive haptic feedback is that the user is likely to be less accurate in mentally tracking the position and orientation of the stick. Whereas the user can track their hand accurately through proprioception, the user is less likely to be able to keep track of the position and orientation of a handheld stick. Therefore, using a handheld stick should allow using greater offsets between the real and virtual objects without the offset being detected by the user. This has the potential to increase the redirection thresholds for passive haptic feedback, which in turn increases the frequency of haptic feedback opportunities. A third potential benefit of touching the environment with a stick and not with one's hand is safety, as failures of the redirection algorithm resulting in unexpected collisions with real world objects will be less dangerous when they involve the stick and not the user's hand.

Driven by these promises, researchers have begun investigating providing haptic feedback through a handheld stick. Early results are encouraging [49] but the haptic redirection algorithms developed are limited to the scenario when the user hits a vertical wall both in the virtual and the real world. Furthermore, prior work requires that the user experience the virtual world by tapping, having to retract the stick after contact is made, before making the next contact with the virtual object.

In this paper we propose a general handheld stick redirection method that allows the user to experience a variety of virtual objects with haptic feedback. The virtual objects are mapped to the *same* simple real world object, i.e., a disk. Furthermore, the user can experience haptic feedback both by tapping or by making continuous contact with the virtual object, as needed, for example, to trace the virtual object's contour. As the user extends the stick to make contact with the virtual object, our method estimates the real and virtual contact points to redirect the virtual stick by translation, rotation, and scaling in order to synchronize the real and virtual contacts. The virtual stick redirection is done in one of two ways: the *FixedHand* method keeps the virtual hand—and therefore the near end of the virtual stick—in place, at its true position, whereas the *FreeHand* method does not impose this constraint. *FixedHand* was designed under the assumption that users are perceptive of any changes to their hand position, so therefore the redirection should be less noticeable if it is not applied to the virtual hand; however, keeping the virtual hand in place concentrates the virtual stick redirection at the far end of the stick where contact is made and therefore on which the user is likely to focus. *Freehand* distributes the stick redirection over the entire stick, which reduces the amount of redirection at the contact point, but which changes the virtual replica of the user's hand.

Fig. 1 illustrates our method. The user is seated in front of a small real world disk anchored to the real world scene (b and d). The user holds the handheld controller to which a stick is attached. The user's hand and the stick are shown in the virtual world. As the user is attempting to tap a virtual 3D star shape (a), our redirection method modifies the virtual stick such that its tip touches the virtual object as the real world stick touches the real disk, see the discrepancy between the real (red) and virtual (blue) sticks in (a). Our redirection method can also operate continually, without resetting, which allows the user to touch the virtual object for a longer period of time, as needed in contour tracing (c). Our redirection method supports large differences in shape between the virtual and real object. Specifically, we have used a thin (2D) real world disk to provide convincing haptic feedback for a variety of virtual objects (e).

We have investigated the effectiveness of our haptic redirection method in a user study ($N = 26$) with two experiments. The first experiment investigates *how far apart* the virtual and real objects can be before the user notices the redirection. The results indicate that the physical object can be anywhere within a 30cm radius of the virtual object for our redirection method to provide haptic feedback without the user noticing. The second experiment investigates *how different the shapes* of the virtual and real objects can be, using an invisible virtual object that the user had to guess by tapping it and by following its contour. Users guessed the correct object out of the lineup for 78% of the time, which shows that our haptic redirection method allows the

user to recognize virtual shapes that are significantly different from the physical object that provides the contact.

In summary, the paper contributes a passive haptic redirection method for a handheld stick that:

- supports a variety of 2D and 3D virtual objects, all mapped to the same real world disk;
- allows for tapping but also for extended contact with the virtual object, as needed, for example, in contour tracing;
- is characterized by large virtual to real offset detection thresholds, as measured in a 2IFC experiment;
- allows a user to guess the shape of an invisible virtual object based on haptic feedback alone, as demonstrated empirically.

2 RELATED WORK

Haptic feedback requires communicating mechanical energy to the user. Researchers have investigated ways of achieving this without making physical contact with the user, e.g., through ultrasound [22]. The haptic feedback methods that do rely on making contact with the user can be categorized in active and passive methods. In active methods, the source of haptic feedback is dynamic, i.e., it provides the energy that generates feedback. Examples include haptic gloves [1, 4], handheld controllers [3, 12, 17, 38], and even computer tablets [45]. In passive methods, the feedback is provided by the reactionary force generated by physical objects in response to user actions [23]. We describe a method for providing haptic feedback to the user of a VR application using a handheld stick with which the user makes contact with real world objects, which falls in the passive haptics category. We rely on redirection to increase the opportunities for passive haptic feedback, by relaxing the requirement for strict alignment between the virtual object and its physical counterpart. As such, we focus the discussion of related work on prior methods for passive haptic feedback (Sect. 2.1), with an emphasis on methods that rely on redirection (Sect. 2.2).

2.1 Passive Haptics

The passive haptic feedback can be provided by handheld props, which are evocative of tools used in the real world. For example, a VR golf application will have the user play the game with a mock golf club [2]. The challenge is to make the same handheld prop as versatile as possible, to be useful in multiple contexts. For this, researchers have developed props that change in shape [32] and weight distribution [42], another example is a palm-size handheld device that allows the user to feel the contact between a virtual handheld stick and virtual objects; the contact is simulated by opening and closing two wings, and by applying pressure on the user's palm [36]. Handheld props such as a haptics cane have been used to assist those visually impaired [46].

The passive haptic feedback can also be provided by tangible objects [18, 20, 21], i.e., physical objects present in the real world hosting the VR application. The approach has the challenge that opportunities for haptic feedback are rather scarce. Although the user will not notice, or accept small differences between the tangible and the virtual object [18], the virtual world is different from the real world, and it is unlikely that there is a real world object aligned with each virtual object the user is about to touch. Researchers have pursued several approaches for increasing the opportunities for haptic feedback from tangible objects.

One approach is to compute a mapping between the real and virtual worlds that maximizes the number of haptic feedback opportunities by finding and leveraging local geometric similarities [11, 33]. A second approach is to devise tangible objects that can impersonate the haptic signature of a large range of (virtual) objects [19].

A third approach is to modify the real world in order to align a real world object with the virtual object the user is about to hit. One example of such encountered-type haptic device (ETHD) relies on a robot to assemble tangible objects at run-time, as needed by the user of a VR application [48]. Another example of ETHD relies on a robotic arm to place a rectangular surface at the anticipated virtual world contact

point [37]. A wider range of contact positions is achieved by replacing the robot with a drone [7, 39]. One concern is safety, as the user does not see the robot arm or the drone and could make contact with them inadvertently. Other limitations are the latency of these mechanical positioning systems, as well as their cost [15]. For a more detailed discussion of ETHD haptics, we refer the reader to a recent survey paper [26].

A fourth approach for increasing the number of passive haptic feedback opportunities relies on redirection, which we discuss in the next section.

2.2 Redirection for Haptic Feedback

Redirection has been first developed to alleviate the size mismatch between large virtual environments and the smaller physical spaces hosting the VR application. Such redirected walking techniques [29] abandon a rigid one to one mapping between the real and virtual worlds to change the users' position and orientation in the virtual world. The goal is to steer the user clear of real world obstacles that do not have a virtual world counterpart and hence cannot be seen by the user. Redirection has been extended to a general human-computer interaction paradigm. Redirection is used in haptics to increase the number and believability of haptic feedback opportunities.

Redirection has been used to increase the versatility of a handheld prop, i.e., by modifying the prop's position and orientation in the virtual world to provide more convincing haptic feedback [27, 35]. For example, a VR grabbing tool that provides haptic feedback for precise virtual object manipulation had its grabbing range increased through redirection [40].

Haptic redirection has been applied to the user's hand [8, 25, 27, 41], for the virtual hand to make contact with the virtual object when the real hand makes contact with a real world object. Detectability thresholds were measured to guide applications in the use of haptic redirection [43]. Like in the case of redirected walking, redirection is less detectable by the user when it is applied during moments of inattention, e.g., when the user blinks or looks in a different direction [28, 44]. The versatility of the grounded haptic feedback can be increased if the user does not make contact with a real world object that happens to be present in the scene, but rather with a prop designed specifically to reduce the distances that have to be covered through redirection [16]. Props that can change shape dynamically are able to simulate contact with a greater variety of objects, at the cost of higher hardware complexity [6]. Other than changing the shape of the real object directly, changing the virtual object can also convince the user, which has been proven by distorting the virtual object to match the available physical object [9, 10].

In order to map several virtual objects to the same stationary physical object, one has to redirect the user's hand such that the same real world grabbing motion be mapped to several different virtual grabbing motions [47]. In addition to manipulating the user's hand trajectory in the virtual world, researchers have also demonstrated the benefits of manipulating the user's virtual fingers to bridge size differences between virtual and real objects that are grabbed by the user [14]. Haptic redirection has also been applied to improve the versatility of a physical simulator for arthroscopic knee surgery training; using VR and by warping the operating field space, the physical simulator can be used with a variety of virtual knee anatomies and pathologies, improving the breadth of training without the expense of multiple physical simulators [34].

Recently researchers have explored the possibility of increasing the availability and believability of haptic feedback by adding one level of indirection between the user and the environment that they explore: instead of touching the environment directly with their hand, users touch the environment with a handheld stick [49]. One advantage is that the stick conveys less information to the user, which, paradoxically, can make the haptic feedback more believable. The user feels what they expect to feel when they touch a real world object with a real world stick, without the missing temperature or texture cues. Second, as they move the stick, the user cannot predict the position of the tip of the stick as accurately as they can predict the position of their own hand, which opens the door for larger redirection thresholds. Finally,

using a stick makes experiencing haptic feedback safer, as inadvertent collisions with the stick do not pose the same risk as inadvertent collisions involving the user's hand. In other words, the stick brings a degree of separation between the user's hand and the tangible object, which attenuates or even filters out incongruencies between the virtual and tangible objects. Prior work confirmed these advantages in the limited context of tapping a virtual wall, with a redirection approach that required for the user to reset their hand position before initiating another contact with a virtual object [49]. The Our paper investigates providing passive haptic feedback to a user exploring the virtual world using a handheld stick. We introduce a redirection method that provides haptic feedback for a variety of virtual objects, without the need to reset, enabling both short, i.e., tapping, and extended contact, as needed, for example, in contour tracing. The new haptic redirection algorithm affords convincing feedback in a wider range of scenarios, which is confirmed in a study where the participant has to leverage the haptic feedback to identify a hidden object.

3 DYNAMIC REDIRECTION OF HANDHELD STICK FOR PASSIVE HAPTIC FEEDBACK

Handheld stick redirection for providing haptic feedback has to be designed to maximize the distance between real and virtual objects that it can bridge without the user noticing. For this, the redirection method should be designed with the following concerns in mind.

- *Continuity.* The virtual stick should be modified gradually, without abrupt jumps in position, orientation, or length, as gradual changes are less noticeable.
- *Synchronization.* The virtual stick should collide with the virtual object precisely at the same time that the real stick collides with the real object. A strict synchronization of the real and virtual contacts is a sine qua non requirement for the believability of the haptic feedback.
- *Extended contact.* The user should be able to investigate a virtual object for extended periods of time, and not just by tapping. Extended contact is necessary for the user to gain insights about the object shape beyond the simple reactionary forced perceived on collision.
- *Use of tip and side of the stick.* The user should be able to investigate the object both with the tip of the handheld stick and with the side of the stick. Whereas the tip of the stick is harder to control and could fall off the object, using the side of the stick allows for a more stable extended investigation of the profile of the object.

Figure 2 gives the pipeline of our method for dynamic redirection of a handheld stick for passive haptic feedback. The real stick is attached to the handheld controller. The position of the real stick tip in the controller's coordinate system is pre-calibrated (all preprocessing steps are shown in grey). This allows tracking the real stick as the user moves it. The geometry and pose of the real world object is also determined in preliminary step using the handheld controller. The geometry of the virtual object and its pose is also known, as it is provided by the VR application.

For each frame, the pose of the real stick is updated based on the tracker data and on the pre-calibrated stick length and orientation with respect to the handheld controller. Then the contact points on the real stick, real object (disk), virtual object, and virtual stick are updated sequentially as described in Sect. 3.2, using the pre-calibrated radius and pose of the real disk, and the application-provided geometry and pose of the virtual object. Finally the virtual stick is scaled, rotated, and translated to redirect it such as the virtual stick and object and the real stick and object make contact at the same time (Sect. 3.3).

3.1 Real World Object Selection

When the user makes contact with an object using a handheld stick we distinguish between two cases: the contact point the tip of the stick, or

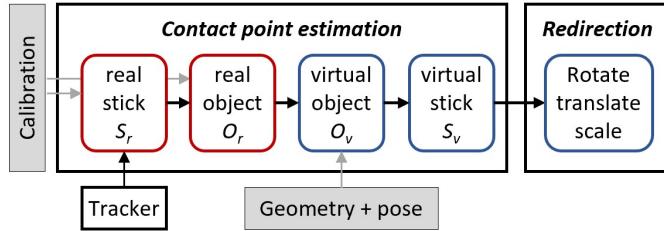


Fig. 2. Pipeline of our dynamic redirection method for passive haptic feedback with a handheld stick. The contact points on the real stick, real object, virtual object, and virtual stick are updated continuously, for every frame, in this order. The contact points are then used to update the length, position, and orientation of the virtual stick, redirecting it to synchronize the virtual and real collisions.

the contact point is on the side of the stick. In the first case, the haptic feedback helps the user probe the distance to the object, i.e., in the z direction. In the second case, no matter where the point is on the side of the stick, the haptic feedback helps the user probe the contour of the object, i.e., in the x and y directions.

With help from redirection, the simplest shape that provides this feedback is a vertical disk. The disk will be able to convey the varying thickness of a 3D virtual object when the stick makes contact with the face of the disk. Indeed, a 3D shape is not necessary since the virtual 3D shape will have a different depth and redirection will still have to hide depth differences between the two 3D shapes. In other words, the disk presents the average depth of the 3D virtual shapes it is called to impersonate.

Similarly, the disk will be able to convey the general contour of the 3D virtual object when the stick makes contact with the contour of the disk. Indeed, a shape with a more complex contour is not necessary since the virtual 3D shape will have a different contour and redirection will still be needed to hide the differences between the two contours. In other words, the disk presents the average contour of the virtual shapes it is called to impersonate.

A further argument in support of choosing a disk as the real world object is the observation that it is harder to hide real object shape discontinuities without virtual object counterparts than it is to hide the absence of real discontinuities to match virtual discontinuities. The reason for this is that the visual feedback provided to the user dominates the haptic feedback so seeing without feeling is preferable to feeling without seeing [13]. We have confirmed this in a pilot study where we used a rectangle as a real world object. When the rectangle was called upon to impersonate a virtual shape with a continuous contour such as a disk, a sphere, an ellipse, or an ellipsoid, users would easily notice the haptic discontinuities at the corners of the rectangle.

Final arguments in favor of the disk are its symmetry which promises uniform detection thresholds in all directions, and its convexity, which facilitates reaching real contour points without inadvertent collisions.

3.2 Contact Point Estimation

There are two pairs of contact points that the redirection method has to update continuously, for every frame. One pair, defines the contact in the virtual world: the contact point on the virtual stick S_v , and the one on the virtual object O_v . Another pair defines the contact in the real world: the contact point on the real stick S_r , and the one on the real disk O_r . Our method updates the four contact points starting with the real contact pair (S_r, O_r) , which then determines the virtual contact pair (S_v, O_v) . This order is dictated by the fact that the redirection is applied to the virtual stick, i.e., the malleable quantity is the virtual and not the real stick. Once the best opportunity for real world contact is established, the virtual contact is determined to guide the user towards seizing this opportunity.

Updating real contact points (S_r, O_r) . The update of the real contact points is illustrated in Fig. 3. When the real stick intersects the plane of the disk (a), S_r is set to the intersection point, and O_r is set to the disk

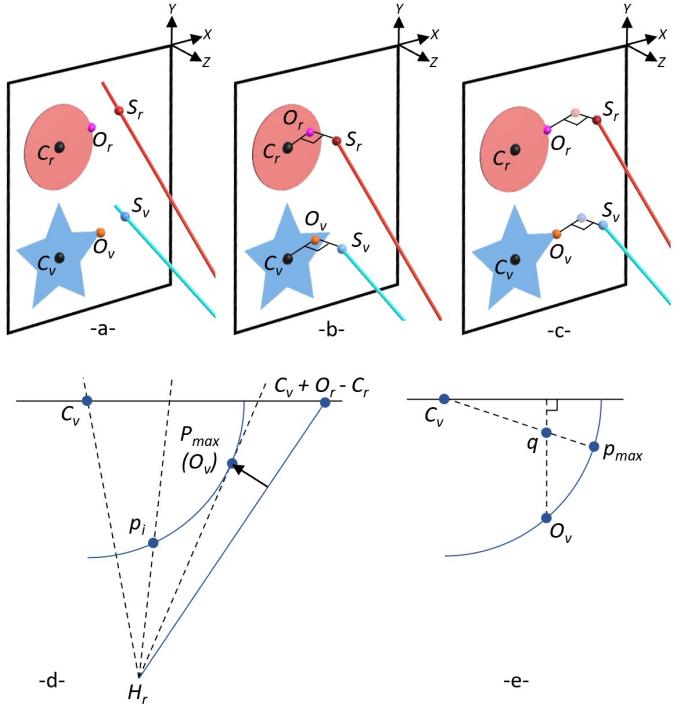


Fig. 3. Contact point estimation. Real stick and object (disk) are shown in red, virtual object and stick are shown in blue. There are three cases: when the real stick intersects the plane of the disk (a), when the far end of the stick projects inside the disk (b) and outside the disk (c). O_v is updated as shown in (d) and (e).

point that is closest to S_r . When the real stick does not intersect the disk plane (b and c), S_r is the far end of the stick. When S_r projects inside the disk (b), O_r is set to the projection point. When S_r projects outside the disk (c), O_r is set to the disk point that is closest to the projection point.

Updating virtual object contact point O_v . The contact point O_v on the virtual object is computed with Algorithm 1. The algorithm first defines the current hitting plane Π through three points: the center of the virtual object C_v , the real world hand position H_r , and a point offset from C_v along the same vector as O_r is offset from C_r (line 2). Fig. 3 (d) and (e) illustrate the computation of O_v in plane Π . The algorithm finds the maximal hitting point p_{max} , from H_r , by considering all edges e_i of the virtual object that intersect Π (lines 3-6). If O_r is on the contour of the disk (i.e., cases (a) and (c)), then O_v is p_{max} (line 9 and (d)). Otherwise (i.e., case (b)), p_{max} is shifted by linear interpolation to q using the fraction f defined by the position of O_r on its radius from C_r (line 11 and (e)). After the shift, O_v is computed as the intersection between the virtual object geometry and the disk plane normal through q (line 12 and (e)).

Updating virtual stick contact point S_v . The contact point S_v with the virtual stick is updated as shown in Algorithm 2. The algorithm proceeds independently for each of the three coordinates of the 3D point S_v (line 1). A coordinate is updated according to one of three cases. The first case handles the situation when S_r at the previous frame was aligned with O_r at the current frame, along dimension i , when S_v is set to be aligned with O_v (line 3). The second case handles post real contact configurations, i.e., the real stick and object contacts are close (in practice we use an ϵ of 2cm) and the distance is increasing (line 4). In this case S_v is set to an offset from C_v equal to the offset between S_r and C_r (line 5), which prevents three problems: the virtual stick remaining locked on the contact point, the virtual stick accelerating too abruptly, and the virtual stick moving in a direction opposite to the real stick. The third case handles configurations when contact is not imminent, and S_v is moved towards the updated virtual object contact point O_{v1} with a velocity commensurate to that which the real contact

Algorithm 1 Virtual object contact point update

Input: real object geometry G_r , real hand position H_r , real object contact point O_r , virtual object geometry G_v
Output: virtual object contact point O_v

- 1: $C_v = G_v.\text{center}$; $C_r = G_r.\text{center}$
- 2: $\Pi = \text{Plane3Points}(C_v, C_v + O_r - C_r, H_r)$
- 3: $p_{\max} = C_v$
- 4: **for** $e_i \in G_v.\text{edges}$ **do**
- 5: **if** $(p_i = \Pi \cap e_i) \wedge (\angle_{C_v, H_r, p} > \angle_{C_v, H_r, p_{\max}})$ **then**
- 6: $p_{\max} = p_i$
- 7: $f = (O_r - C_r) / G_r.\text{radius}$
- 8: **if** $f = 1.0$ **then**
- 9: $O_v = p_{\max}$
- 10: **else**
- 11: $q = C_v + (p_{\max} - C_v)f$
- 12: $O_v = G_v \cap (q, q + G_r.\text{normal})$
- 13: **return** O_v

points move towards each other (lines 7-8).

Our algorithm for updating the virtual stick contact point (Algorithm 2) has the following desirable properties. First, it uses a short history of a single frame, which avoids locking onto a contact point early. The user is free to change their mind regarding where contact is made. Second, the algorithm updates the three coordinates independently, which moves the virtual stick in a similar way to the real stick, conveying to the user a sense that they are actually controlling the virtual stick with fine granularity. Fig. 4 (a) illustrates the different motion of the virtual stick compared to the real stick with a naive algorithm that aggressively reduces the distance between contact points. Indeed, as the user moves the real stick from S_{r0} to S_{r1} , the virtual stick moves in an almost perpendicular direction, i.e., from S_{v0} to S_{v1} . This large discrepancy between the real and virtual stick motions is likely to be detected by the user, and also to be found objectionable. On the other hand, our algorithm (b) does not move S_v in x, as the real stick does not move in the x direction, providing a closer alignment between the real and virtual stick motions.

3.3 Handheld Stick Redirection

S_v is now a known 3D point on the virtual stick. The remaining degrees of freedom for the virtual stick are determined with one of two methods. The first method, *FixedHand*, keeps the virtual hand collocated with the real hand (Fig. 5a). This determines the near end H_v of the virtual stick and a single degree of freedom remains, i.e., the length of the virtual stick. When the real stick contact point S_r is the tip of the real stick, i.e., cases (b) and (c) in Fig. 3, then the tip of the virtual stick is set to S_v , which determines the length of the stick. When the real stick contact point S_r is along the real stick, i.e., case (a) in Fig. 3, the length of the virtual stick l_v is set for the contact point S_v to divide the virtual stick in the same ratio as S_r divides the real stick of length l_r (Equation 1).

Algorithm 2 Virtual stick contact point update

Input: real and virtual object centers C_r and C_v ; real and virtual object and stick contact points O_r, S_r, O_v, S_v ; indices 0 and 1 indicate previous and updated values
Output: updated virtual stick contact point S_{v1}

- 1: **for** $i \in \{x, y, z\}$ of S_{v1} **do**
- 2: **if** $S_{r0.i} = O_{r1.i}$ **then**
- 3: $S_{v1.i} = O_{v1.i}$
- 4: **else if** $|O_{r1.i} - S_{r0.i}| < |S_{r1.i} - O_{r1.i}| < \epsilon$ **then**
- 5: $S_{v1.i} = C_v.i + S_{r1.i} - C_r.i$
- 6: **else**
- 7: $f = (S_{r1.i} - S_{r0.i}) / (O_{r1.i} - S_{r0.i})$
- 8: $S_{v1.i} = S_{v0.i} + (O_{v1.i} - S_{v0.i}) * f$
- 9: **return** S_{v1}

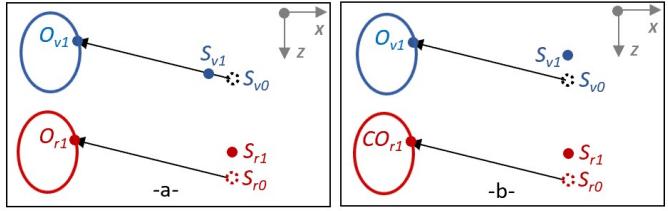


Fig. 4. Updating the virtual stick contact point S_v with a naive (a) and with our (b) algorithm. Virtual entities are shown in blue and real entities are shown in red. The naive algorithm results in a motion of the virtual stick that is different from that of the real stick. Our algorithm conveys to the user a sense that they are controlling the virtual stick, which moves in a way similar to the real stick.

$$\|S_v H_v\| / l_v = \|S_r H_r\| / l_r \quad (1)$$

The second method, *FreeHand* allows for the translation of the virtual hand away from the position of the real hand (Fig. 5b). The position of the virtual hand H_v is computed by translating the position of the real hand H_r by half the offset between S_v and S_r . Once H_v is known, the length of the virtual stick is determined like for *FixedHand*.

4 EVALUATION

We have conducted a user study with the approval of our Institutional Review Board.

Participants. We have recruited $N = 26$ participants with the following demographic data. The age range was between 20 and 30 years, with an average of 24.38. There were 6 women. 1 participant never used a VR application before, 9 once, 13 occasionally, and 3 frequently. Participants were involved in a single session between 40 and 50 minutes long.

Implementation and setup. We implemented our *FixedHand* and *FreeHand* redirection methods in Unity 3D (2021.3.4) [5], and the VR application was ported to a Meta Quest 1 VR headset [3]. A 50cm real bamboo stick with a diameter of 0.6cm and a weight of 15 grams was attached to the handheld controller using a custom 3D-printed plastic bracket. The real stick was calibrated with respect to the controller and it was assumed to be rigid during the experiments. The participant was seated approximately 1m away from a vertical disk with a 15cm diameter. The disk was attached to a stable optical table such the participant had access to the front of the disk as well as to its sides (Fig. 1 (b) and (d)). The participant was allowed to use their dominant hand. The participant was seated such that their shoulder corresponding to their dominant hand was directly in front of the disk. The system rendered the virtual hand and stick (blue in Fig. 1), but not their real world counterparts.

Each participant performed two experiments.

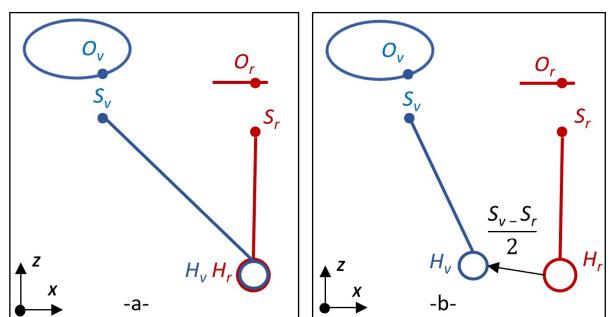


Fig. 5. *FixedHand* (a) and *FreeHand* (b) stick redirection.

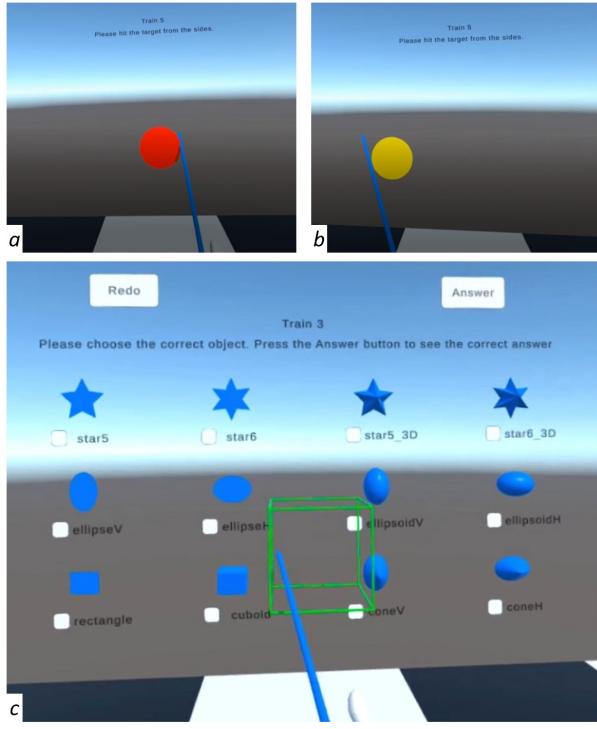


Fig. 6. Experiment 1 (a and b) and experiment 2 (c) frames. For experiment 1, the user is asked to hit pairs of virtual disks, one at the time, one with and one without redirection, and then to indicate the disk where redirection was applied. For experiment 2, the user is asked to investigate with the handheld stick an invisible virtual object located inside the green wireframe box, and to pick it out of the lineup of 12 possible objects.

4.1 Experiment 1: Redirection Detection Thresholds

The first experiment measured how far the real and virtual objects can be without the participant noticing the redirection.

Experimental Design. The redirection thresholds were measured with a two-interval forced choice (2IFC) design. The participant was shown two virtual disks, one at the time (Fig. 6 a and b). Exactly one of the virtual disks was offset with respect to the real disk. The participant was asked to tap each disk, from the left or right side, with the virtual stick. Participants could choose whether to hit the disk from the left or from the right. The disks had different colors chosen from the set {green, red, yellow, black}. Each disk is shown for one more second after it is tapped by the user and then it disappears automatically.

After the second disk disappears, the participant was asked "For which disk did the stick NOT move as expected?". The possible answers were shown side by side, the disks were referred to using their colors, with the first disk always left and the second disk always right. The participant selected the answer by aiming the virtual laser pointer at the desired disk and by clicking a controller button. Since the answers were shown above the workspace where the virtual disks appeared, the participant had to raise the stick to select the answer, and therefore the participant would always begin a new trial with the stick above the virtual disk to be tapped. This is a neutral position for the side tapping that the participant had to perform. The participant was forced to choose one of the two disks before being able to proceed to the next trial. There were 21 trials per condition, i.e., seven values for the offset $\epsilon \in \{-30\text{cm}, -20\text{cm}, -10\text{cm}, 0\text{cm}, 10\text{cm}, 20\text{cm}, 30\text{cm}\}$, each repeated three times. The experiment used a counterbalanced design.

Conditions. We measured four pairs of detection thresholds τ : one for each of the two redirection methods *FixedHand* and *FreeHand*, and for each of the vertical and horizontal offset direction. For each pair, one threshold corresponds to negative translations, i.e., to the left or down, and one threshold corresponds to positive translations, i.e., to the right or up. We used a within-subject design, with each participant per-

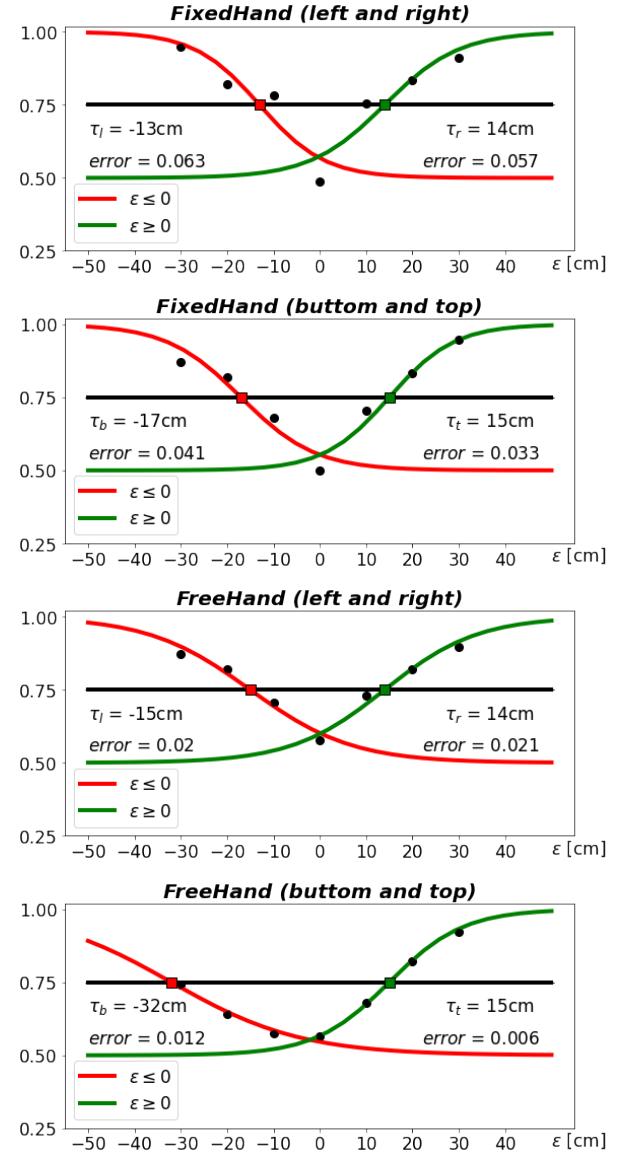


Fig. 7. Detection thresholds for the two redirection methods: τ_l and τ_r for left and right offsets, and τ_b and τ_t for bottom and top offsets. For each graph, the y axis is the percentage of correct answers and the x axis is the offset between the real and virtual objects.

forming trials for each condition. In total, each participant performed 84 trials, 21 for each of the four conditions (i.e., pairs of thresholds).

Training session. Before the actual experiment, each participant performed 28 practice trials: one for each of the seven threshold values and for each of the conditions.

Data collection and analysis. Each participant did 112 trials in total and the average completion times was 19.41 ± 3.37 minutes. We recorded the correctness of each of the participant's answers. The answer is correct if the participant correctly identifies the offset disk. For each participant, the answers were averaged for each offset value, which results in seven data points per participant and per condition. The data points were used to fit a psychometric sigmoid function (Equation 2).

$$\Psi(\epsilon; \alpha, \beta, \gamma) = \gamma + (1 - \gamma) \Sigma(\epsilon; \alpha, \beta) \quad (2)$$

$$\Sigma(\epsilon; \alpha, \beta) = \frac{1}{1 + e^{\delta \frac{\epsilon - \alpha}{\beta}}} \quad (3)$$

FixedHand						FixedHand vs FreeHand	
$\tau_l - \tau_r$		$\tau_l - \tau_b$		$\tau_l - \tau_t$		$\tau_l - \tau_l$	
Norm.	Wilcoxon	Norm.	t-test	Norm.	Wilcoxon	Norm.	t-test
0.017	Z = -0.47, p = 0.638	0.495	t = -0.829, p = 0.415	0.015	Z = -0.296, p = 0.767	0.065	t = 0.257, p = 0.799
	$\tau_r - \tau_b$		$\tau_r - \tau_l$		$\tau_b - \tau_t$		$\tau_r - \tau_r$
Norm.	Wilcoxon	Norm.	t-test	Norm.	t-test	Norm.	t-test
0.032	Z = -0.114, p = 0.909	0.211	t = 0.07, p = 0.945	0.804	t = -0.253, p = 0.802	0.052	t = -0.015, p = 0.988
FreeHand							
	$\tau_l - \tau_r$		$\tau_l - \tau_b$		$\tau_l - \tau_t$		$\tau_b - \tau_b$
Norm.	t-test	Norm.	Wilcoxon	Norm.	t-test	Norm.	t-test
0.195	t = 0.830, p = 0.415	0.005	Z = -3.586, p < 0.001	0.631	t = -1.094, p = 0.284	0.257	t = -3.597, p = 0.001
	$\tau_r - \tau_b$		$\tau_r - \tau_l$		$\tau_b - \tau_t$		$\tau_t - \tau_t$
Norm.	t-test	Norm.	t-test	Norm.	t-test	Norm.	t-test
0.517	t = -2.724, p = 0.012	0.826	t = -0.245, p = 0.786	0.154	t = -2.925, p = 0.007	0.303	t = -0.503, p = 0.620

Table 1. Statistical comparison of pairs of detection thresholds, within and across the two methods. Normal data (normality p-value > 0.05) was analyzed with a t-test and non-normal data with a Wilcoxon test. The only significant differences ($p < 0.05$) are for the pairs involving the τ_b of FreeHand (red font).

Parameter γ gives the lower range of the psychometric function. For our 2IFC experiment $\gamma = 0.5$ since when redirection is not detectable the answer correctness percentage matches chance behavior. In other words, if the user cannot detect the stimulus, i.e. if the user cannot identify the virtual disk where redirection is applied, they will choose one disk randomly, which will be the correct answer 50% of the time. δ is either +1 or -1, to model an increasing or decreasing psychometric function. In our case, $\delta = -1$ for negative offsets and $\delta = +1$ for positive offsets. The fit optimized parameters α and β to minimize the mean square distance from the curve to the data points. α positions the sigmoid horizontally and β scales the sigmoid laterally and represents the slope inversely. The detection thresholds are determined at the intersection between the sigmoid and the 75% answer correctness line.

In addition to fitting the psychometric function, we have also compared the detection thresholds for the two methods, i.e., *FixedHand* and *FreeHand*. As required by our within-subjects design, we used a paired t-test when the data showed a normal distribution, and a non-parametric Wilcoxon test [30] for data that violated the normality assumption. Normality was investigated with the Shapiro-Wilk test [31].

Results and discussion.

Fig. 7 shows the four pairs of detection thresholds measured by our experiment: τ_l and τ_r for left and right, and τ_b and τ_t for top and bottom, for each of the two redirection methods. The functions were fit to the pooled data of all participants. In all cases the answer correctness was close to 50% for an offset ε of 0cm. The sigmoid fits the data well. The fitting error is measured vertically, i.e., on the y axis, so its units are percentage of correct answers. The largest fitting error is for *FixedHand left*, which means that for an offset of $\tau_l = -13\text{cm}$ the expected percentage of correct answers is $75\% \pm 6.3\%$. The tightest fit is for *FreeHand top*, where for an offset of $\tau_t = 15\text{cm}$ the percentage of correct answers is $75\% \pm 0.6\%$.

The detection thresholds are large and of uniform value, except for *FreeHand* τ_b which, at -32cm, is twice as large as the other thresholds. We explain this exception with the observation that when the virtual disk is low in front of the participant, the frame projection of the virtual stick is small. This prevents the participant from noticing the redirection that, in the case of *FreeHand*, acts like a translation along the stick. *FixedHand* scales the virtual stick which is more noticeable even when the stick has a small frame projection.

The statistical analysis of the difference between detection thresholds is given in Table 1. We fit a psychometric function to the answers of each participant and used the thresholds of individual participants to determine the significance levels of differences between conditions. However, due to the small number of repetitions, the resolution of the psychometric data is coarse and only gives approximate comparisons among the conditions. The table confirms that the τ values are similar for all offset directions and all methods, i.e., the significance value p is larger than 0.05, except for the bottom offset of the *FreeHand*

redirection method. The absence of significant differences between left and right offsets confirms that there is no dominant-hand bias.

In summary, experiment 1 indicates that participants cannot reliably detect redirection for offsets in the [-15cm, +15cm] range.

4.2 Experiment 2: Shape Perception Through Haptics

The second experiment investigates the quality of the haptic feedback provided by the *FreeHand* method by asking the user to guess the shape of an invisible virtual object based on the haptic feedback received.

Experimental Design. The participant is asked to guess the shape of an invisible object inside a green wireframe box using the handheld stick (Fig. 6 c). The participant has to pick the correct object out of a gallery of 12 possible objects: a 2D and a 3D five-pointed star, a 2D and a 3D six-pointed star, a vertical and a horizontal 2D ellipse, a vertical and a horizontal ellipsoid, a 2D rectangle, a cuboid, a horizontal and a vertical elliptic cone. The objects were selected to allow investigating the fidelity of the haptic feedback along multiple dimensions: the direction of the shape change (xy versus xyz, i.e., 2D versus 3D), orientation (horizontal versus vertical), geometric detail (five- versus six-pointed stars), and geometric similarity (ellipsoid versus elliptic cone). The participant reaches with the handheld stick to make contact with the face of the invisible object and to trace its contour. The participant can take as long as needed before deciding which virtual object is hidden. The gallery of possible objects is visible as the participant investigates the hidden virtual object, so the participant can use it to rule out hypotheses. There are 24 trials per participant, with each of the 12 hidden objects being used twice, in counterbalanced manner.

An important aspect of the experiment design is to understand the relative contribution of the visual and haptic feedback provided to participants to assist them in guessing the invisible object. Although the object is not shown, the virtual stick is visualized and its motion contributes to revealing the shape of the hidden object. However, when the participant makes contact with the object using the side of the stick, the contact point is not visualized. In other words, the user cannot visually trace the contact point to receive a strong indication of the shape of the object. It is true that the user sees the tip of the virtual stick, but for side contact the tip of the stick is not the contact point. Indeed, the locus of the tip positions is not a close approximation of the object contour Fig. 8, hence the participant does not rely exclusively on visual feedback to recognize the object. Furthermore, the haptic feedback provided by the (real) disk is important to the hidden object recognition as it blocks the real and therefore the virtual stick at the contour of the object, not letting the stick cut through the object. Implementing the object recognition task without the disk would amount to the user not being able to trace the contour of the hidden object interactively, and the redirection algorithm would have to essentially engage in moving the stick automatically on the contour of the hidden object, without or against the motions of the user.

	star5	star5_3D	star6	star6_3D	rectangle	cuboid	ellipseV	ellipseH	ellipsoidV	ellipsoidH	coneV	coneH
star5	79	17	4	0	0	0	0	0	0	0	0	0
star5_3D	8	92	0	0	0	0	0	0	0	0	0	0
star6	4	2	83	12	0	0	0	0	0	0	0	0
star6_3D	2	8	0	90	0	0	0	0	0	0	0	0
rectangle	4	0	2	0	58	19	2	13	0	2	0	0
cuboid	2	0	0	0	12	79	2	4	0	2	0	0
ellipseV	0	0	0	0	2	0	87	4	8	0	0	0
ellipseH	0	0	0	0	0	0	0	98	0	2	0	0
ellipsoidV	0	0	0	0	0	2	12	2	67	4	12	2
ellipsoidH	0	0	0	0	0	0	0	10	0	75	0	15
coneV	0	0	0	0	0	2	6	0	27	0	63	2
coneH	0	0	0	0	0	0	0	4	2	25	2	67

Table 2. Results of the shape perception experiment. A table element T_{ij} , with $i > 1$ and $j > 1$ indicates the percentage of times object T_{i1} was guessed as object T_{1j} . The percentage of correct answers are given on the main diagonal, which indicates how many times an object was correctly identified as itself. The table cells are shaded in green, with darker shades corresponding to larger percentages.

Data collection and analysis. For each trial, we record the answer provided by the participant, and not just its correctness. In addition to the percentage of correct answers, this also allows judging which object is mistaken for which.

Results and discussion. The results are given in Table 2. Each row starts with the hidden virtual object and then each column provides the frequency of the answers provided by participants. For example, based on row 1, the 2D five-pointed star was guessed correctly 79% of the time, 17% of the time it was mistaken for a 3D five-pointed star, and 4% of the time it was mistaken for the 2D six-pointed star. The best performance was recorded for the horizontal ellipse, which was recognized correctly 98% of the time, and it was mistaken for the horizontal ellipsoid 2% of the time.

The worst performance was recorded for the rectangle, which was nonetheless recognized correctly 58% of the time. One reason the rectangle is challenging is because, unlike the real disk, it has corners, which have a strong haptic signature. However, the stars also have corners, and stars are recognized at a high rate. A possible explanation is that the fewer and less salient corners of the rectangle make it harder to discern from the ellipse, with which it was confused 13% of the times. Another reason the rectangle is challenging is that it is difficult to discern its shape along the z axis, i.e., perpendicular to the rectangle face. Indeed, the participant cannot easily investigate the thickness of

the rectangle to disambiguate it from its 3D counterpart, i.e., the cuboid. As the participant moves the virtual stick from the rectangle contour to the face of the rectangle, the jump is abrupt and it is hard to judge the thickness of the rectangle. Therefore the rectangle was confused with the cuboid for 19% of the time, and, vice versa, the cuboid was confused with the rectangle for 12% of the time. Unlike the cuboid, the other 3D objects have a less abrupt change in depth (i.e., along the z axis) which allows distinguishing them more easily from their 2D counterpart. For example, the 3D six-pointed star was never confused with the 2D six-pointed star.

The largest numbers in the table *not* on the main diagonal, i.e., the largest absolute errors, indicate the confusion between the ellipsoid and the cone of the same orientation. The vertical cone was mistaken for the vertical ellipsoid for 27% of the times, and the horizontal cone was mistaken for the horizontal ellipsoid for 25% of the time. For a haptic investigation, an elliptic cone is hard to distinguish from an ellipsoid, as the user has to remember to trace the cone over the precise center of its phase to detect the apex, and all other trajectories of the tip of the virtual stick are plausible in the context of an ellipse.

Regarding the orientation of the invisible object, a horizontal object is rarely confused with its vertical counterpart. For example, the horizontal and vertical ellipses were only confused with one another for 0% and 4% of the time, the horizontal and vertical ellipsoids for 0% and 4%, and the horizontal and vertical elliptic cones for 2% and 2%. Regarding shape detail, the 2D stars were confused with one another only 4% and 4% of the time, and the 3D stars 0% and 8%.

In summary, the haptic feedback provided by our method allowed participants to identify the hidden virtual object correctly for 78% of the time. The method has proven to be robust to shape complexity, orientation, similarity, and is capable of providing haptic feedback along all three cardinal directions.

5 CONCLUSIONS, LIMITATIONS, AND FUTURE WORK

We have demonstrated the versatility of our haptic redirection method by showing how complex virtual shapes can be mapped to a simple real world disk. By probing the space occupied by an invisible virtual object with a handheld stick, our study participants identified the hidden virtual object with remarkable accuracy.

In our work we have converged on a disk, which proved to be quite versatile at impersonating the haptic signature of many objects. The disk had difficulties conveying the abrupt change in z of the cuboid, thus the research problem of designing a physical prop that optimizes versatility remains open. A promising direction of future work is to generate the optimal physical prop given a set of virtual objects that the application is keen on having the user experience with haptic feedback.

Our haptic redirection method was developed for providing haptic feedback for small objects in front of the user. By factoring in the redirection thresholds measured by experiment 1, it is likely that the 14cm diameter disk is expanded to cover a 30cm x 30cm x 30cm region. In our current work we did not explore the threshold for vertical tapping,

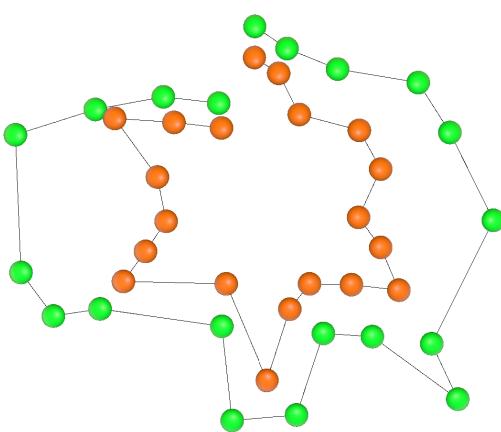


Fig. 8. Trajectory of virtual object contact point (orange) and of the virtual stick tip (green) during a hidden 2D six-pointed star recognition task. Whereas the contact point does reveal the contour of the object, the contact point occurs on the side of the stick and is not conveyed visually to the participant. The trajectory of the tip does not closely match the contour of the object, hence the participant cannot rely exclusively on visual feedback to recognize the object.

which future work could explore. We hypothesize that tapping from the top downwards could yield larger detectability thresholds since the stick moves faster, under gravity, and since the stick tends to occlude the contact point. Implementing haptic displays with larger volume can be pursued by using our method in the context of encountered-type haptic devices where the object providing feedback moves to meet the user's handheld stick. For example, when the object providing feedback is mounted on a robotic arm, using our redirection method could lower the robot motion distance and velocity requirements, reducing cost and increasing safety. Achieving haptic displays of larger volume can also be pursued by designing a collection off physical objects that can best cover a given volume. Our disk can serve as a first "pixel" towards the design of larger passive haptic displays.

In our current work we have always assumed that the physical object providing the haptic feedback is exactly in front of the user. First, this introduces the possibility of bias in the threshold measurement of experiment 1, since the virtual disk with the redirection was always the one offset with respect to the user. The results of the experiment do not indicate that participants relied on this fact to consistently identify the disk with redirection. Furthermore, this limitation of the study can only result in smaller and not larger thresholds, hence the values reported are safe for use in applications. The second consequence of only testing in scenarios where the physical object is exactly in front of the user is that the thresholds for offset physical objects are yet to be discovered in future work.

Furthermore, our current work deals with stationary virtual worlds. Future work could examine providing haptic feedback as the user makes contact with moving virtual objects. A first step is to indicate how much of the virtual object motion can be absorbed by redirection. Subsequent efforts could investigate using a robotic arm to provide the needed change in position of the physical prop providing the haptic feedback. Furthermore, a robotic arm could also allow the user to change the velocity of a moving virtual object in agreement with its mass, enabling for example remote science laboratories with tangible interfaces.

Our work provides haptic feedback for outside-looking-in exploration where the user is seated and examines a virtual object. Future work could explore scanning the real world scene and to use it to provide haptic feedback, as the user moves through the virtual environment, exploring it in inside-looking-out fashion.

Our work advocates conveying a physical dimension to the virtual world through a handheld stick, based on the assumption that providing less but accurate information to the user is preferable to providing more but occasionally egregiously wrong information. Users trust what they see and extrapolate the basic haptic feedback they *perceive* to match the much more complex haptic feedback they *expect*. Like in the context of underwater swimming where users move their hands slowly to remain in agreement with their avatar that experiences drag from the virtual water [24], we have observed our users altering the trajectory of their hands to move the virtual stick around the points of the star in a way that maintains contact with the complex shape. Such user cooperation can be exploited in future work to simulate surfaces with various textures, for example by having the tip of the virtual stick vibrate as it crosses surface ridges.

Finally, our work measured *detection* thresholds whereas many applications would be well served by redirection levels at the *acceptability* thresholds, in line with the hope of using technology to not just match, but also surpass reality.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their help with improving this manuscript, and the members of the Computer Graphics and Visualization Laboratory of the Computer Science Department of Purdue University for their feedback. This work was supported in part by the United States National Science Foundation through awards 2212200 and 2219842, by a gift from Meta, and by the Purdue University Computer Science Department.

REFERENCES

- [1] Haptx vr gloves. <https://haptx.com/>. Accessed: 2022-09-22.
- [2] Oculus golf. <https://www.oculus.com/experiences/quest/2412327085529357/>. Accessed: 2022-10-13.
- [3] Oculus headset. <https://www.oculus.com/>. Accessed: 2022-09-29.
- [4] touchdiver. <https://www.weart.it/>. Accessed: 2022-10-8.
- [5] Unity 3d engine. <https://unity.com/>. Accessed: 2022-09-29.
- [6] P. Abtahi and S. Follmer. Visuo-haptic illusions for improving the perceived performance of shape displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2018.
- [7] P. Abtahi, B. Landry, J. Yang, M. Pavone, S. Follmer, and J. A. Landay. Beyond the force: Using quadcopters to appropriate objects and the environment for haptics in virtual reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2019.
- [8] M. Azmandian, M. Hancock, H. Benko, E. Ofek, and A. D. Wilson. 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*, pp. 1968–1979, 2016.
- [9] Y. Ban, T. Kajinami, T. Narumi, T. Tanikawa, and M. Hirose. Modifying an identified angle of edged shapes using pseudo-haptic effects. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*, pp. 25–36. Springer, 2012.
- [10] Y. Ban, T. Narumi, T. Tanikawa, and M. Hirose. Displaying shapes with various types of surfaces using visuo-haptic interaction. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology*, pp. 191–196, 2014.
- [11] L. Beever and N. W. John. Leveled sr: A substitutional reality level design workflow. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 130–138. IEEE, 2022.
- [12] H. Benko, C. Holz, M. Sinclair, and E. Ofek. Normaltouch and texture-touch: High-fidelity 3d haptic shape rendering on handheld virtual reality controllers. In *Proceedings of the 29th annual symposium on user interface software and technology*, pp. 717–728, 2016.
- [13] C. C. Berger, M. Gonzalez-Franco, E. Ofek, and K. Hinckley. The uncanny valley of haptics. *Science Robotics*, 3(17):eaar7010, 2018.
- [14] J. Bergström, A. Mottelson, and J. Knibbe. Resized grasping in vr: Estimating thresholds for object discrimination. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, pp. 1175–1183, 2019.
- [15] E. Bouzbib and G. Bailly. "let's meet and work it out": Understanding and mitigating encountered-type of haptic devices failure modes in vr. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 360–369. IEEE, 2022.
- [16] L.-P. Cheng, E. Ofek, C. Holz, H. Benko, and A. D. Wilson. 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*, pp. 3718–3728, 2017.
- [17] I. Choi, E. Ofek, H. Benko, M. Sinclair, and C. Holz. Claw: A multifunctional handheld haptic controller for grasping, touching, and triggering in virtual reality. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–13, 2018.
- [18] X. de Tinguy, C. Pacchierotti, M. Emily, M. Chevalier, A. Guignardat, M. Guillaudeux, C. Six, A. Lécuyer, and M. Marchal. How different tangible and virtual objects can be while still feeling the same? In *2019 IEEE World Haptics Conference (WHC)*, pp. 580–585. IEEE, 2019.
- [19] X. de Tinguy, C. Pacchierotti, M. Marchal, and A. Lécuyer. Toward universal tangible objects: Optimizing haptic pinching sensations in 3d interaction. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 321–330. IEEE, 2019.
- [20] D. Harley, A. P. Tarun, D. Germinario, and A. Mazalek. Tangible vr: Diegetic tangible objects for virtual reality narratives. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, pp. 1253–1263, 2017.
- [21] A. Hettiarachchi and D. Wigdor. Annexing reality: Enabling opportunistic use of everyday objects as tangible proxies in augmented reality. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 1957–1967, 2016.
- [22] T. Howard, M. Marchal, A. Lécuyer, and C. Pacchierotti. Pumah: Pan-tilt ultrasound mid-air haptics for larger interaction workspace in virtual reality. *IEEE transactions on haptics*, 13(1):38–44, 2019.
- [23] B. E. Insko. *Passive haptics significantly enhances virtual environments*. The University of North Carolina at Chapel Hill, 2001.
- [24] H. Kang, G. Lee, and J. Han. Visual manipulation for underwater drag force perception in immersive virtual environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 38–46. IEEE, 2019.

2019.

[25] L. Kohli. *Redirected touching*. PhD thesis, The University of North Carolina at Chapel Hill, 2013.

[26] V. R. Mercado, M. Marchal, and A. Lécuyer. “haptics on-demand”: A survey on encountered-type haptic displays. *IEEE Transactions on Haptics*, 14(3):449–464, 2021.

[27] N. C. Nilsson, A. Zenner, and A. L. Simeone. Propping up virtual reality with haptic proxies. *IEEE Computer Graphics and Applications*, 41(5):104–112, 2021.

[28] C. Patras, M. Cibulskis, and N. C. Nilsson. Body warping versus change blindness remapping: A comparison of two approaches to repurposing haptic proxies for virtual reality. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 205–212. IEEE, 2022.

[29] S. Razzaque. *Redirected walking*. The University of North Carolina at Chapel Hill, 2005.

[30] D. Rey and M. Neuhäuser. *Wilcoxon-Signed-Rank Test*, pp. 1658–1659. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. doi: 10.1007/978-3-642-04898-2_616

[31] S. S. Shapiro and M. B. Wilk. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3/4):591–611, 1965.

[32] J. Shigeyama, T. Hashimoto, S. Yoshida, T. Narumi, T. Tanikawa, and M. Hirose. Transcalibur: A weight shifting virtual reality controller for 2d shape rendering based on computational perception model. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, 2019.

[33] A. L. Simeone, E. Velloso, and H. Gellersen. Substitutional reality: Using the physical environment to design virtual reality experiences. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3307–3316, 2015.

[34] J. Spillmann, S. Tuchschmid, and M. Harders. Adaptive space warping to enhance passive haptics in an arthroscopy surgical simulator. *IEEE transactions on visualization and computer graphics*, 19(4):626–633, 2013.

[35] P. L. Strandholt, O. A. Dogaru, N. C. Nilsson, R. Nordahl, and S. Serafin. Knock on wood: Combining redirected touching and physical props for tool-based interaction in virtual reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020.

[36] Y. Sun, S. Yoshida, T. Narumi, and M. Hirose. Pacapa: A handheld vr device for rendering size, shape, and stiffness of virtual objects in tool-based interactions. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pp. 1–12, 2019.

[37] E. Vonach, C. Gatterer, and H. Kaufmann. Vrrobot: Robot actuated props in an infinite virtual environment. In *2017 IEEE Virtual Reality (VR)*, pp. 74–83. IEEE, 2017.

[38] E. Whitmire, H. Benko, C. Holz, E. Ofek, and M. Sinclair. Haptic revolver: Touch, shear, texture, and shape rendering on a reconfigurable virtual reality controller. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–12, 2018.

[39] K. Yamaguchi, G. Kato, Y. Kuroda, K. Kiyokawa, and H. Takemura. A non-grounded and encountered-type haptic display using a drone. In *Proceedings of the 2016 Symposium on Spatial User Interaction*, pp. 43–46, 2016.

[40] J. Yang, H. Horii, A. Thayer, and R. Ballagas. Vr grabbers: Ungrounded haptic retargeting for precision grabbing tools. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, pp. 889–899, 2018.

[41] A. Zenner, H. M. Kriegler, and A. Krüger. Hart-the virtual reality hand redirection toolkit. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–7, 2021.

[42] A. Zenner and A. Krüger. Shifty: A weight-shifting dynamic passive haptic proxy to enhance object perception in virtual reality. *IEEE transactions on visualization and computer graphics*, 23(4):1285–1294, 2017.

[43] A. Zenner and A. Krüger. Estimating detection thresholds for desktop-scale hand redirection in virtual reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 47–55. IEEE, 2019.

[44] A. Zenner, K. P. Regitz, and A. Krüger. Blink-suppressed hand redirection. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 75–84. IEEE, 2021.

[45] L. Zhao, Y. Liu, D. Ye, Z. Ma, and W. Song. Implementation and evaluation of touch-based interaction using electrovibration haptic feedback in virtual environments. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 239–247. IEEE, 2020.

[46] Y. Zhao, C. L. Bennett, H. Benko, E. Cutrell, C. Holz, M. R. Morris, and M. Sinclair. Enabling people with visual impairments to navigate virtual reality with a haptic and auditory cane simulation. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–14, 2018.

[47] Y. Zhao and S. Follmer. 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*, pp. 1–12, 2018.

[48] Y. Zhao, L. H. Kim, Y. Wang, M. Le Goc, and S. Follmer. Robotic assembly of haptic proxy objects for tangible interaction and virtual reality. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, pp. 82–91, 2017.

[49] Y. Zhou and V. Popescu. Tapping with a handheld stick in vr: Redirection detection thresholds for passive haptic feedback. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 83–92. IEEE, 2022.