”Let’s Meet and Work it Out”: Understanding and Mitigating Encountered-Type of Haptic Devices Failure Modes in VR

Elodie Bouzbib, Gilles Bailly

"Let's Meet and Work it Out": Understanding and Mitigating Encountered-Type of Haptic Devices Failure Modes in VR

Elodie Bouzbib[1,2]*
[1] INRIA, Université de Rennes, CNRS, IRISA
Rennes, France

Gilles Bailly[2]†
[2] CNRS, Sorbonne Université, ISIR, ISCD
Paris, France

ABSTRACT

Encountered-type of Haptic devices (ETHD) are robotic interfaces physically overlaying virtual counterparts prior to a user interaction in Virtual Reality. They theoretically reliably provide haptics in Virtual environments, yet they raise several intrinsic design challenges to properly display rich haptic feedback and interactions in VR applications. In this paper, we use a Failure Mode and Effects Analysis (FMEA) approach to identify, organise and analyse the failure modes and their causes in the different stages of an ETHD scenario and highlight appropriate solutions from the literature to mitigate them. We help justify these interfaces' lack of deployment, to ultimately identify guidelines for future ETHD designers.

Keywords: Encountered-type of Haptic Devices, Haptics, Virtual Reality, Design, FMEA, Theoretical Framework, Robotic Graphics

1 INTRODUCTION

Enriching Virtual reality (VR) experiences using Haptics has become a timely topic in the last few years, especially with the expansion of affordable Head-Mounted Displays (HMDs) and VR applications - such as gaming or industry training.

Most of current haptic technologies for VR applications are grounded desktop interfaces, or ungrounded controllers (custom controllers, wearables, exoskeletons, handheld devices), which both require the user to hold an interface continuously. An emerging class of technologies are Encountered-Type of Haptic Devices (ETHD). ETHD are robotic devices physically encountering the users to overlay a virtual counterpart through haptic and force-feedback in VR. Their main advantage is therefore to leave the users "unencumbered" of any contraption (such as wearables, controllers etc) while providing a high level of visuo-haptic consistency and enabling various interactions (e.g. navigation, exploration, manipulation [10]). Yet, while their concept is promising for providing Haptics in VR, current interfaces face many design and implementation challenges, which stall them in a prototyping phase and justify their lack of deployment. The aim of this paper is to identify ETHD intrinsic challenges.

We propose to use a *FMEA* - Failure Modes and Effects Analysis approach in these regards. This "Failure mode" approach is a common protocol in Industries Product Design and Quality Control [50]. We argue that it enables to identify, analyse and emphasize the main ETHD challenges. Using the FMEA approach, we first (1) depict Encountered-type of Haptic Devices scenarios (current usability), to (2) highlight their principal functions, (3) describe their potential failure modes, causes and effects, and (4) provide solutions from the literature to mitigate them. We believe that using this method provides the necessary step back for designers to improve the ETHD usability and facilitate their deployment, as it helps to identify and refine these interfaces' global specifications (hardware and software).

*e-mail: elodie.bouzbib@inria.fr
†e-mail: gilles.bailly@sorbonne-universite.fr

2 BACKGROUND: ETHD

In this section, we discuss the origin, the definition and current methods to design and evaluate ETHDs.

2.1 Origin

Krueger introduced in 1993 a parallel between artificial and physical realities [57], explaining how *"humans are mobile, unencumbered or tethered creatures"*, and concluded with a technical question: *"Is unencumbered artificial reality not possible?"*. The same year, McNeely was coming up with *Robotic Graphics* [37], a novel concept for robotic devices for force-feedback in VR, encountering the user to enable interactions. Robotic Graphics are at the origin of Encountered-Type of Haptic Devices (ETHD).

2.2 Definitions

McNeely's conceptual Robotic Graphics are split into two distinct categories:

– *Robotic Shape Displays - RSD*. "A robot is present that can **reach** any location on the virtual desktop with an end effector [...] When the system **anticipates** contact, it orders the robot" to **display** it for the **interaction** to occur.

– *Roboxels*[1]. "cellular robots that dynamically **configure** themselves into the **desired** shape and size, lock together and simulate the desired object." They can be used for exploration or manipulation tasks, or synchronised with the users gestures to enable edition tasks [10].

There are six important keywords in these definitions: *anticipate* contact, *reach* location, *display/configure* the *desired* object, and *interaction*. We will use them in the Section 3.5 to define the main functions and sub-functions of ETHDs.

ETHD. The core concept of ETHD is anchored in the Robotic Graphics principle, but the terminologies differ depending on the communities (Robotics, VR, HCI, Haptics). The first ETHD implementation was introduced in the VR community as *WYSIWYF - What You See Is What You Feel* displays: "what the user can see via a visual interface is consistent with what he/she can feel through a haptic interface. The user's hand **"encounters"** the haptic device exactly when his/her hand touches a virtual object in the scene" (Figure 1) [62]. This definition introduces the term "encounters", which currently is the 'E' of ETHDs.

[1]Roboxel is standing for "robotic volume element".

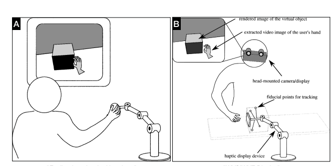

Figure 1: Illustrations from [61]: (A) Feeling here but seeing there; (B) ETHD (also referred to as a WYSIWYF Display).

Ever since the WYSIWYF, various terminologies emerged: "shape display" [59], "dynamic shape display" [47], "encountered-type haptic display" [20,53], "encountered-type of haptic device" [9], and depend on the community designing them. Yet, all of these terms actually depict the same concept, from a different perspective. In these regards, the haptics and VR community provide a focus on the provided haptic feedback, with the generation of "physical characteristics, such as shape and rigidity, of three-dimensional (3D) virtual objects" [53] "directly explored with his or her bare-hands" [20,59]. The HCI community focuses on enabling the "sensation of voluntarily eliciting haptic feedback with the environment at a proper time and location" [41].

In this article, ETHD refers to **Encountered-Type of Haptic Device** and we propose the following definition: "ETHDs are robotic devices encountering the unencumbered users to physically overlay a virtual Object Of Interest (OOI) through haptic and force-feedback in VR". An object of interest (OOI) is an interactable object from the virtual environment. It refers to the previous *desired* object keyword.

2.3 Design and Evaluation

We observed an increasing number of ETHD prototypes [41] these last years, reflecting the importance of enhancing haptic feedback with unencumbered technologies in VR. While these prototypes investigate various promising design points, it remains unclear how to guide their design and evaluation. We summarise the main methods.

Usability study. Most of the proposed ETHD prototypes have been tested empirically with human participants. The user study generally aims at highlighting the benefits of haptic feedback vs. non haptic feedback [27], the degree of immersion/realism [51] or enjoyment [2,9]. These studies are essential to demonstrate the potentials of ETHDs, but do not provide an holistic view of the challenges to address to deploy these technologies. More specifically, many technical issues are acknowledged in these evaluations (e.g. high latency [27,51,54], light embedded masses [27,51]) and considered as *out of the scope*. Several essential features of these prototypes are thus not used (e.g. real-time displacements [51]) or even simulated by human operators to not impair the usability during the user study. The virtual environments can also be constrained from the beginning to ensure the global interface usability.

Technical study. ETHDs often rely on advanced robotic technologies with multiple parameters. Technical studies aim at measuring the key factors such as the robot speed [19], accuracy [2], response time [27] or robustness [9]. They also aim at determining the optimal parameters for a given task. For instance, [9] determine optimal parameters to increase the target detection delay and enable the ETHD to reach its position prior to the user. While technical studies often seem to highlight a potential solution, they do not explicitly mention their associated failure, its cause or effect. One objective of our approach is to gather and organize all these aspects *(failures)* as well as to discuss software and hardware solutions to mitigate them.

Taxonomies. A recent taxonomy provides an overview of the available ETHD technologies [41]; which can be used by novice designers to identify the most adequate solutions. Authors distinguish *grounded solutions* (Robotic arm [33] or Fixed platform [28]), or *ungrounded solutions* (drone [59], mobile platform [60], on-demand handheld [15,25]).

Criteria. Recently, Nilsson et al. highlighted two criteria to consider when designing ETHDs, and more generally VR technologies using physical props [42] in order to provide a high level of physicality. These two criteria are *co-location* and *similarity*. Co-location corresponds to the accurate physical overlaying of virtual objects; Similarity corresponds to the consistency between the virtual and physical object haptic properties. This work confirms that ETHDs have the potential to provide both co-location and similarity and emphasises high-level challenges. The same year, Bouzbib et al. highlighted the importance of the task (e.g. navigation, manipulation, exploration) and the type of scenario (deterministic vs. non-deterministic) on the choice of the ETHD technologies [10] and called to consider more systematically the following criteria: speed, accuracy, robustness and safety. Nonetheless these papers do not discuss the concrete ETHD challenges, their causes and the associated mitigation solutions, to enable these criteria.

In this paper, we focus on the intrinsic challenges designers face when designing and evaluating ETHDs, we emphasize their causes and effects. We believe it can help designers identify and better understand them, and facilitate the deployment of these interfaces.

3 Approach: Failure Modes and Effects Analysis

We propose to use the FMEA approach to analyse ETHD challenges.

3.1 Definition

FMEA - *Failure Modes and Effects Analysis* - is a common approach in Industries and Product design. " "Failure modes" refer to the ways in which something might fail. Failures are any errors or defects, especially ones that affect the user, and can be potential or actual. "Effects analysis" refers to studying the consequences of those failures" [5,50]. FMEA enables to better understand how to achieve a successful design and aims at enhancing a product reliability: it helps defining product specifications, as it analyses the main principal *functions* (or sub-functions) and requirements the product must achieve.

3.2 Procedure

In order to use a Failure modes identification process, we first need to identify the product main function. This main function is then depicted in *sub-functions*, identifying the "tasks" it *must* perform, and therefore translating it into product requirements. Among the functions, we then have a Troubleshooting phase where we identify all the potential failures, their causes and their effects.

The failures effects are usually rated in terms of severity (e.g. an unsafe interface colliding with a high speed into a user will cause injuries) and probability of occurrence (e.g. a slow robot is likely to not reach its target prior to interaction). The combination of these criteria enables designers to identify and rate the most critical failures to mitigate in a criticality matrix, along with their detectability.

3.3 FMEA & ETHDs

We propose to use the FMEA approach to study ETHDs as transposing approaches from industry to research can foster the deployment "beyond the prototype" [24,31]. We believe this approach can provide the necessary step back to improve the ETHD usability, by analysing and discussing the different failures designers must take into account when designing and implementing ETHDs.

As we do not focus on a single design but on current literature designs and their associated mitigation solutions, we thus do not provide the criticality information - which is interface-dependent but can be implemented to compare interfaces sharing common properties. However, expert designers focusing on their own interfaces can obviously exploit failures from our survey to draw severity/probabilities features and mitigate their most critical failures.

3.4 Global Aim

The aim is to identify the potential failures in ETHDs, and to provide a framework highlighting these interfaces' mitigation solutions, implicitly promoted in the literature. The FMEA approach was previously used in its entirety (criticality etc) to define specifications of a VR system [49], yet in this paper we do not critically analyse a given application but provide a groundwork for future ETHD designers to get acquainted with future potential challenges and to be aware of the current solutions to solve them. This paper provides an overview of the ETHD literature through an original and practical approach.

3.5 Requirements identification

We built upon the current ETHD literature and our experience in ETHD design to emphasize these interfaces' challenges and to discuss their potential solutions. Analysing ETHDs from their core definitions provides their principal functions; their associated hardware and software implementations from the literature reflect their requirements and help describing their sub-functions.

We build on the keywords from the previous definitions (Section 2.2) to depict a classic scenario: a user in a VR environment, wearing a Head Mounted Display and free to interact with any object of interest with no regards to the scenario's progress. This is referred to as a *non-deterministic* [9], or an an *unscripted experience* [13][2]:

> *"Matt enters a virtual room. He sees a table, on a corner of the room, on which are displayed a cylinder and a cube; and a cupboard, on which is displayed a pyramid. (a) He visually navigates through the environment, and (b) intends to interact with the cylinder. (c) The ETHD captures this intention, and (d) moves towards the corresponding cylinder position in the physical world - to physically overlay it in the virtual one. (e) It changes its configuration to simulate a cylinder primitive. In the meantime, (f) Matt reaches for the cylinder. (g) He finally interacts with it with a great visuo-haptic consistency, both spatially and temporally."*

Figure 2: An Encountered-Type of Haptic Device scenario: 1. Intentions, 2. Displacement & Reconfiguration, 3. Interaction.

We understand from this scenario, that for the interaction to occur, the main function of the robot is to be displaced *prior* to interaction. It can be summarized through 3 steps: **1. Intentions**, **2. Displacement and 2.bis Reconfiguration**, **3. Interaction** (see Figure 2), and reflecting the main keywords of Robotic Graphics. The sub-functions and requirements in this scenario are[3]:

1. *Intentions* The user navigates and intends to interact with an object *(a, b & c)*.
   - The robot *must* capture the user intention (prediction algorithm success rate);
   - The robot *must* predict which object of interest to overlay (prediction algorithm resolution).
2. *Displacement* The user and the robot both move towards the chosen object of interest *(d)*.
   - The robot *must* reach the chosen object prior to the user (speed requirement).
   - The robot *must* reach the chosen object accurately (accuracy requirement) and reliably (precision requirement).
   - The robot *must* avoid the user during its displacement (safety requirement).

2.*bis* *Reconfiguration* The robot changes its configuration *(e)*. This step is a type of displacement - occurring *within* the robot.
   - The robot *must* display the adequate prop, adequate end-effector or *must* reconfigure itself with the adequate shape (resolution requirement).

3. *Interaction* The user now interacts with the robot *(f & g)*.
   - The robot *must* display the adequate object properties (haptic transparency requirement) and *must* enable their associated exploratory procedures.

[2]Otherwise, the experience is constrained, defined as "scenario-based".
[3]We attach the associated letter from the scenario to each Step in *italics*.

4 Failures in the Intentions - Step 1

Predicting the users next objects of interest can be perceived from an arena to a desktop scale. In an arena, the algorithm is required to anticipate the future object of interest within a walking delay, while at a desktop scale, the algorithm is required to anticipate it within a reach-to-touch phase. The prediction algorithm performance is usually rated in terms of prediction success rates.

4.1 Failure Causes

The **algorithm delay** to predict the users' next object is a main cause of failure. They might be too short for the algorithm to predict the targets. They are actually a field of study, for instance in Unscripted retargeting [13], they are measured in order to determine at which stage of the movement a target can be acquired accurately (81% accuracy of correct target acquisition at 65% of the movement).

The algorithm success rate is also a function of its resolution: an algorithm can be successful whenever two objects are a meter apart, yet when they are 10cm apart it is unable to capture the intention correctly. The **parameters** that can change a prediction algorithm resolution are: (a) the number of virtual objects of interest; (b) the distance between them; (c) their respective sizes. The more objects available for interactions, the better resolution the algorithm must display. Hence, proposing a large number of objects and/or of small sizes and/or close to each other can cause failures in the prediction.

Moreover, as ETHD are often designed to enable bare-hands and unencumbered interactions, a **lack of information** from the users can complicate the prediction, as the input for prediction models should rely on the users behaviours. For instance, intention prediction models can involve gaze and hand coordination, which hence require both an eye and a hand-tracker [7].

The **algorithm inputs** can also cause a failure: we can potentially imagine an algorithm relying on users' gestures or behaviours. The algorithm would require enough robustness to cater for unexpected movements or brutal gestures.

4.2 Failure Effects

The effect of this failure is simple: if the algorithm does not predict the future target, the robot cannot physically overlay it, and the interaction cannot occur. When the prediction is correct but the delay is too short, this results in a failure in the displacement of the interface and ultimately in the overlaying of the virtual object with a physical prop.

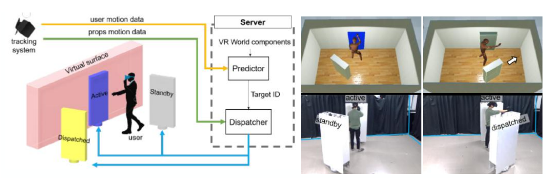

Figure 3: ZoomWalls [60] Active, Standby and Dispatched walls.

4.3 Solutions for Mitigation

An obvious solution for mitigating a failure in the prediction algorithm would be to rely on **scenario-based experience**, such as with the *Beyond the force* drone experience [2]: the robot does not need to anticipate the users' behaviours, as the full experience is scripted. Instead of determining which object of interest is about to be interacted with - which thus also removes this intention prediction step, interfaces can also **follow the users directly** and stop whenever the user touches them. This requires sufficient speed and safety measures to avoid any collision. These interfaces are referred to as *Encounter-type* of haptic displays [20]. They differ from ETHD (Encounter**ed**-type of haptic devices as they do not anticipate the users movements but follow them from a small distance. Theoretically, when their speed is sufficient, these interfaces can cater for unexpected movements from the users as they remain at a constant distance from them and within their vicinity.

In the same regards, the robot can also follow the users to **overlay the closest object** from the users' vicinity. This is for instance implemented with the Snake Charmer [4] robotic arm.

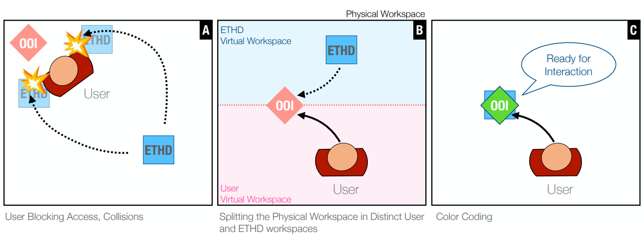

Figure 4: A. The ETHD cannot reach the Object of Interest because the user is blocking the access. This causes collisions and safe trajectories cannot be generated. B. The physical workspace is split so the user and the ETHD do not collide with each other whilst displacing. If the user arrives before the ETHD, we can consider coupling this with other techniques such as the C. "color coding": when the interface is at the exact location, the OOI's color changes to inform the user that the interaction can occur.

Increasing the number of robots to overlay the closest objects is a good alternative to refining an intention prediction model. This was implemented in VRRobot, which uses three robotic arms around the user [56], and with Roomshift, where mobile robots move objects around the VR arena [51].

Zoomwalls employs a swarm of mobile robots to overlay virtual walls with physical ones (see Figure 3), and assigns a different role to each of them. It distinguishes the *Active*, *Standby* and *Dispatched* ones. All of the Standby robots are following the user. When an interaction decision is made, the closest Standby robot becomes the Dispatched one. Once it reaches the interaction position, it then becomes Active It only considers users with a walking speed below $0.4m/s$ [60] to ensure the algorithm delay is long enough, and **ask the users to slow down** otherwise. This compromise is a result from a two-variable technical evaluation: number of swarms and walking speed. Too many swarms (> 3) shows drawbacks as they collide with each other; not enough swarms causes the virtual objects not to be overlaid in time. Also, and no matter the number of swarms, a higher walking speed ($> 0.4m/s$) causes a failure in the identification of the chosen object of interest.

Another solution is for designers to **add prior probabilities** to non-deterministic experiences [9]: it is still unscripted, yet the algorithm decision phase is facilitated. For instance, if a basketball hoop and a ball were available for interaction, one can assume that the ball would be interacted with first. Instead of being equally available, the objects are weighted accordingly with the probability to be interacted with; the interface moves at the centroid of these weighted objects' positions. This mitigation solution was shown to increase the success rate in overlaying a virtual object in time for interaction from 80% to more than 93% in non-deterministic scenarios simulations up to 4 distractors [9]. Finally, another alternative is to **decrease the number of available virtual objects**, and/or **increase their sizes or their spacing**.

5 Failures in the Displacement - Step 2

Once the intention has been predicted and the Object of interest (OOI) determined, the ETHD has eventually a position command to reach. For the ETHD to be successful in its displacements, it is required to move towards it, whilst simultaneously avoiding the user and reaching it prior to him/her.

5.1 Failure Causes

The failures in the interface displacement can be **speed**, **accuracy** and **safety** related. Indeed, the interface trajectory generation must (a) avoid the user and any unwanted collision and (b) reach the target accurately (c) prior to the user.

Consequently, the causes for failures can be from a **collision with the user**, justifying that the dynamic trajectories are not generated correctly or do not take into account the users' displacements. The interface can potentially displace itself safely around the user but miss the target because of a lack of accuracy, speed, or because of an **uncontrolled deceleration** or even **oscillations** [48].

Regarding safety, we can also depict failures in the **reachability of the interface** (see Figure 4 - A): either the object of interest to overlay is too far from the interface's workspace centre (at its boundaries) and it becomes inaccessible for the robot - because the user is blocking its access, by being too close from the object - or because the interface cannot reach the target without colliding with the user.

5.2 Failure Effects

The most important criterion in the design of an experiment blending users with robotised interfaces is to ensure their safety. Therefore, an experiment with unsafe trajectory generations will not be able to be tested and even less deployed.

A lack of speed or accuracy can result in the interface not being available prior to the user, which compromises the interaction. A failure will cause what we define as a *spatial mismatch* (or co-location issue [42]): the interface is not physically overlaying its virtual counterpart because of its spatial position. Ultimately, the users would not be able to interact with their virtual environment with haptic feedback.

5.3 Solutions for Mitigation

Safety-wise, a good solution to mitigate the risks is to **run simulations** where the user would not be hurt. For instance, designers can record users doing the experiment without haptic feedback, to eventually run simulations with an avatar (simulated user) and real interfaces movements, such as with CoVR's implementation [9]. Many different algorithms ensuring safe environments with dynamic movements of both the user and the interface are currently available. A solution is to employ algorithms usually developed for swarm interfaces, and to **perceive the user as another interface**, which must be avoided. The interfaces aim to reach the same goal whilst avoiding collisions with each other: indeed, the user and the interface share the same space [32]. The global idea is to extract the positions and speeds of each interface, to determine its future positions and generate trajectories accordingly, oscillation and collision free. This principle is known as the ***Velocity Obstacle*** [18], but has been improved as Reciprocal Velocity Obstacle [55] and Hybrid Reciprocal Velocity Obstacle [48]. **Collision avoidance algorithms** using artificial potential fields have also been developed for teleoperation and manipulators [29, 30]: we can consider artificially representing the user as a high potential obstacle the robot must avoid.

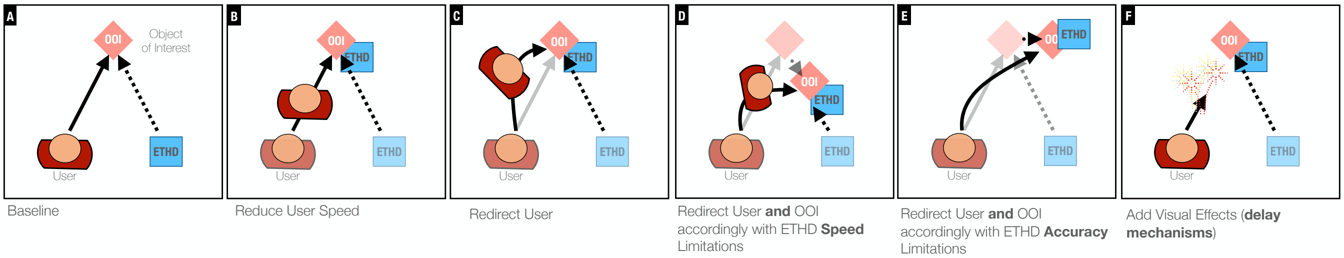

Figure 5: Solutions for Mitigating Displacement failures from Scenario A, the user and ETHD are equidistant from the Object of Interest (OOI). B - The user speed is artificially reduced so the ETHD arrives prior to the user. C - The user trajectory is altered to give spare time for the interface to reach the OOI. D - The user and the OOI position are redirected, to cater for ETHD speed limitations. E - The user and the OOI position are redirected, to cater for ETHD accuracy limitations. F - Visual effects (e.g. fireworks) are added to delay the user, make him stop and give spare time for the interface to place itself.

The previous techniques consider a shared space between the user and the robotised interface. Yet, in order to avoid collisions during the interface displacement, we can also consider **distinguishing the user and the ETHD workspace** (Figure 4-B). The schematic drawing shows a top view of a VR arena, though we can consider this approach by working on different heights as well. This can be considered with drones, being in the air and thus offering interaction opportunities at various heights.

Speed and Accuracy-wise, when the user and ETHD are equidistant from the Object of interest - OOI (Figure 5 - A), this means the ETHD displacement speed must be higher than the user's one. In ZoomWalls [60], the user's speed is reduced by **adding a disturbing noise** (from the movie Predator) in the users headphones when it is above $0.4m/s$. Otherwise, the user trajectory can also be altered *visually*, using **redirection techniques** such as the *Redirected Walking* [43]. Therefore, it avoids the interface trajectory, whilst enabling it to keep the shortest pathway towards the target (Figure 5 - C).

Many ETHDs suffer from accuracy and speed issues - mostly ungrounded ones such as mobile platforms or drones. Redirecting techniques are hence used once again as **dynamic retargeting**: the user's future target is estimated, as well as the interaction time, based on a Jerk model [19]. This estimation time then helps to predict the future position of the ETHD which suffers from speed limitations; the object of interest as well as the user are then redirected for a physical contact to occur (see Figure 5 - D). This also enables to perform experiences in *unconstrained virtual arenas* [19]. Similarly, and as displayed in Figure 5 - E, ETHD such as drones [2] employ this **dynamic retargeting** for the users to physically encounter the ETHD despite its accuracy limitations. Theoretically, even a high error in accuracy ($7cm$) could be tackled using this technique [2], which justifies this mitigation solution robustness and its current deployment in proofs-of-concept.

A Wizard-of-Oz implementation of Robotic Graphics called *TurkDeck* - "Human actuators" are transporting props around the users, instead of a robotised interface - demonstrated techniques to avoid both safety and accuracy/speed issues [12]. They add visual effects to act as **delay mechanisms** (Figure 5 - F), and give some spare time for the human actuators to place themselves adequately.

In the same regards, visual aids can be added to ask the user to wait for the object to be ready for interaction. A **color code** eventually informs the user that the interaction can occur (see Figure 4 - C) [2, 39].

As mentioned in the previous subsection, we suggest the designer **adapt the virtual environment**: while the accuracy of the algorithm could be improved by modifying the number of OOIs, their sizes or spacing, designers can anticipate their interfaces' issues and add visual effects with no regards of its current capabilities. This can also be useful whenever the interface shows no repeatable/precision hardware capabilities (see Figure 6).

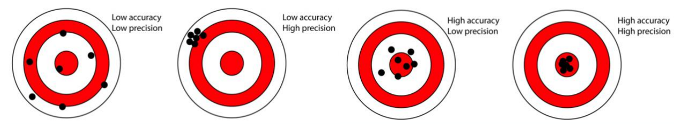

Figure 6: Accuracy vs Precision [14].

Moreover, if the scene design shows objects of interest outside of the interface workspace, which is called a *reachability issue* (see Figure 7 - A) [19], the designer can **establish reachability planes** from the beginning on the experiment design and only offer interaction opportunities within this workspace. In H-Wall [33] for instance, per-planes reachability maps of a Kuka robotic arm with an end-effector were simulated prior to the design of the user experience (Figure 7 - B). Virtual objects were then to be placed within this workspace.

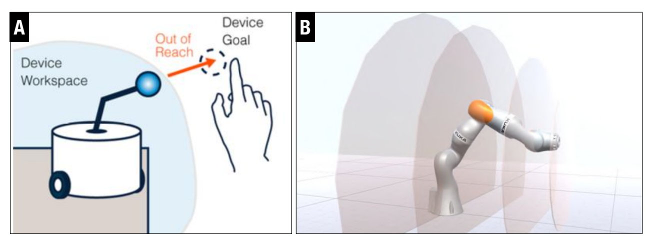

Figure 7: A. Schematic of an ETHD having reachability issues: the device goal is out of reach (from [19]). B. Four per-plane reachability maps of a Kuka robot, from [33].

6 Failures in the Reconfiguration - Step 2.bis

For the interaction to occur, the ETHD either provides the user with a real object, modifies its end-effector or reconfigures itself (Figure 8). A modification of end-effector consists in changing the interactable prop (*between* various objects) while a reconfiguration consist of changing its shape (*within* the object). The end-effector can also be referred to as a *SAD - Shape Approximation Device* [26]. It can be composed of various edges and shape primitives [26], textures or objects [4].

6.1 Failure Causes

An ETHD displacing multiple objects can **display the wrong object** for interaction, yet this is once again can be perceived as a failure in the interface's displacement/accuracy (see previous subsection).

The wrong primitive or object can be displayed or simulated - when using shape-changing devices such as Figure 8 - A [46, 63], 2.5D Tabletops [47] (Figure 8 - B) or Roboxels such as [52, 64] (Figure 8 - C). This can be caused by the **interfaces resolution**.

Beyond the interfaces' resolution and depending on the tasks users are to be performing, a failure can also occur as a **lack of physical coverage**: these interfaces usually physically overlay a part of the virtual object, but not the object in its entirety. For instance, the multi-finger interface in Figure 8 - A is sufficient to explore an object shape through three fingers, but it is not able to cover the palm or the hand in its entirety in a manipulation task.

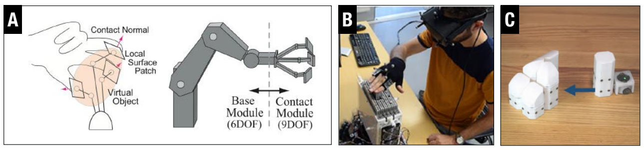

Figure 8: Interfaces reconfiguring themselves for interaction. A. Shape-Changing End-Effector for multi-fingertips, from [46, 63]. B. 2.5D Tabletop, from [47]. C. Reconfigurable robotic elements (roboxels), from [64].

6.2 Failure Effects

We define a failure in the end-effector reconfiguration as a *shape mismatch* (or similarity [42]): the interface is not physically overlaying its virtual counterpart because of its shape. This can result in a *semantic violation*: it is a discrepancy which violates the human body semantics and is heavily rejected by users [44]. It consequently creates a discomfort due to a discrepancy in the visual expectations and the associated haptic feedback.

6.3 Solutions for Mitigation

A first solution to mitigate this effect is to **limit the users tasks, objects of interest variety or shape complexities** within the VR scene. McNeely already suggested in these regards:

> *"RSD would work best in VR scenarios where the objects are of fixed size and appear repetitively, for example, interacting with a virtual radio's layout of knobs, switches and buttons. It is felt that a large number of useful applications, in manufacturing design, and design verification, are amenable to this approach with existing technology."* [37]

McNeely hence evokes using *<**virtual:physical**> **mappings*** (Figure 9): as a thorough 1:1 mapping is more complex to build, a physical object can for instance overlay multiple virtual ones [22, 40].

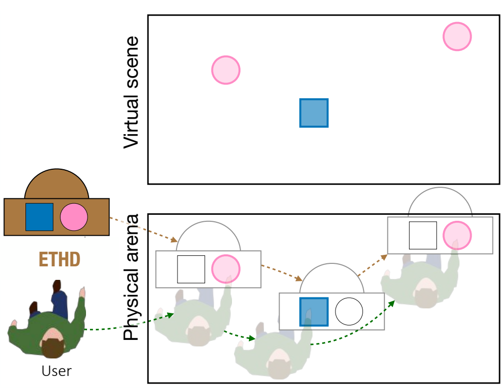

Figure 9: A robotic shape display enabling a 1:1 (blue cube) and a 2:1 (pink ball) *<virtual:physical> mappings*.

Advances in **Pseudo-Haptics**[4] demonstrated that objects of approximately similar primitives can be used to simulate the same object, without altering the user's perception.

This hence also simplifies the use of the previously suggested mappings *shape-wise*: a simple cylinder can either represent a cylinder, a cone, or even a sand glass [6]. Similarly, **redirection and pseudo-haptic techniques** can be used to enable the exploration of complex objects using simple props [64].

Finally, for shape-changing devices, designers can also **alter their virtual scene to match their devices resolution**. This can be perceived as a limitation in the usability of these devices. For instance, ShapeShift [47] and the device from [63] do not show the same level of details to represent a ball, as seen in Figure 10. ShapeShift aimed for a 1.25mm resolution - based on two-point tactile acuity from [11] - but chose to display a 7mm resolution for simplicity purposes; while the end-effectors (Figure 8 - A, Figure 10 - B) display 30mm contact modules.

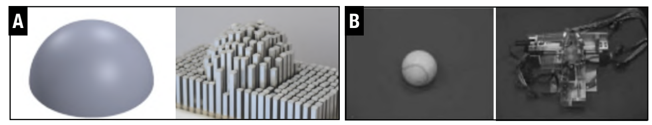

Figure 10: A. High-resolution pin array representing half a ball to explore [47]; B. Contact modules representing a ball through three finger contacts [63].

7 Failures in the Interaction - Step 3

In this section, we focus on the Interaction Step (Step #3) of the ETHD classic scenario (Figure 2 - 3): when the contact does occur and the haptic feedback is provided.

For our analysis, we consider the Lederman associated Object properties and Exploratory procedures [36] illustrated in Table 1. An "Exploratory Procedure" is defined as "stereotyped movement pattern having certain characteristics that are invariant and others that are highly typical. It needs not to correspond to a particular configuration of the hand, a fixed pressure, or a particular end-effector" (see Table 1).

Object Property	Exploratory Procedure
Substance-related properties	
Texture	Lateral Motion
Hardness	Pressure
Temperature	Static contact
Weight	Unsupported Holding
Structure-related properties	
Weight	Unsupported holding
Volume	Enclosure, contour following
Global shape	Enclosure
Exact shape	Contour following

Table 1: Object Properties & associated "Exploratory Procedures" [36].

We exploit these exploratory procedures to analyse the different failure modes *perception-wise*: we assume that a *semantic violation* might occur and damage the haptic experience in VR when they are not correctly simulated.

This section might seem to share some similarities with the previous ones (e.g. semantic violations). However, the perspective is different and provides a complementary picture. Indeed, the previous sections are *designer-oriented*: if the designer failures occur (e.g. an interface does not displace itself accurately or does not reconfigure itself), the user will not be able to interact - this alters the

[4]Pseudo-Haptics leverage the users visual dominance over haptics to alter their perception.

whole ETHD functioning. In contrast, this section is *user-oriented*: the interface succeeded in its prior requirements (e.g. accurate displacement and reconfiguration), yet the user perception is flawed during interaction, through the way he interacts or the procedures he performs (e.g. the user touches the wrong texture).

7.1 Failure Causes

We distinguish the Substance-related Properties from the Structure-related ones (Table 1).

7.1.1 Objects' Substance-related Properties

Going through the different exploratory procedures in Table 1, we can start with the *"Lateral motion"*. A failure can for instance be perceived in the infinite exploration of a wall. Haptically speaking, this motion results in a **tactile slippage** of the skin. Yet, some interfaces enable this infinite exploration whilst moving the interface accordingly with the user's displacements ([4, 60]): the slippage cannot occur.

A failure in the applied *"pressure"* corresponds to a discrepancy in the users' **perceived stiffness** through kinesthetic feedback. This can be translated as an ETHD robustness requirement. Indeed, while robustness can be perceived as a *designer hardware requirement* (e.g. in [10, 17]), it is in fact only required in the third step of the ETHD classic scenario ("Interaction" and user perception, Figure 2 - 3). For instance, if a user is to lean or push on a wall, the ETHD is required to handle this pressure and to react accordingly (e.g. stay still). Yet, this is a common failure with ungrounded solutions such as mobile robots or drones [1], which struggle to compensate for the users applied forces in order to replicate the expected hardness.

For *temperature*-related failures, a study demonstrated that when heat is visually stimulated in an object (a teacup with a fuming beverage for instance), users tend to interact from a cooler location over the object (e.g. the teacup handle) [8]. Yet, if they do interact over the teacup base, they might be expecting a hotter temperature. There is generally a **lack of I/O capabilities** in ETHDs (e.g. combining the ETHD with additional heat cells).

For a user to perceive a correct *weight*, the object of interest is required to be held whilst being unsupported. A common failure of this procedure is known as the **lack of haptic transparency**. The fully transparent haptic device is "an imaginary massless and infinitely rigid stick" [21], which enables the exploration and manipulation of the virtual object without any inertia or friction effect. This failure hence occurs whenever the users are not enabled to perform "a free object manipulation". It can be identified in [2] for instance: when the drone is active, its thrust adds a supplementary weight, and friction effect when holding a prop; when the drone is inactive, its weight will be added to all of the objects attached to it in an unsupported holding.

7.1.2 Objects' Structure-related Properties

Weight in the structural properties of an object is different from its absolute weight (in the substance-related properties): it corresponds to the **inertia** one can perceive when manipulating an object. We can identify this failure with on-demand handhelds [15, 35], where the manipulated object's absolute weight is adequate, yet it generates an inertia in its displacement. Prior to holding the apple, the on-demand handheld provides an unexpected kinesthetic feedback, which alters the perception.

Regarding *enclosure and contour following* for volumes and shapes (Figure 11 - A), we can find a failure for exploration and manipulation tasks (defined in [10]). For instance, it includes users **touching the edge** of a shape approximation device while exploring an object, as the object's enclosure is not fully available (Figure 11 - B): the user explores the shoe at a given location, yet if he moves towards the shoe curvature, a discrepancy will occur.

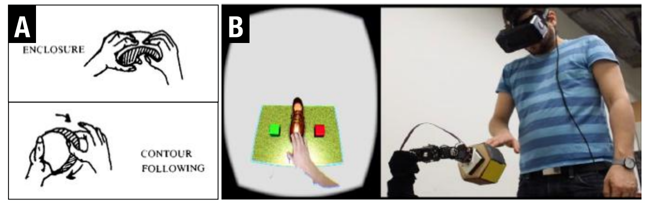

Figure 11: A. Schematics of Enclosure and Contour Following from [36]. B. A User exploring a shoe material with Snake Charmer [4]: if the user moves, he will perceive the edges of the Shape-approximation device; the shoe is not available interaction-wise for Enclosure or Contour following Exploratory procedures.

7.2 Failure Effects

All of the previously listed failures result in a discrepancy from the users expected haptic feedback (in textures, forces, weights, temperatures). For instance, tactile slippage usually enables the user to haptically distinguish textures [16]. Thus, the Lateral motion failure will impact the texture perception. Similarly, perceived stiffness and haptic transparency will respectively impact the hardness and weight perception during the experience.

Structural discrepancies in the object properties (weight, volume, shape-wise) can also potentially mitigate the user experience, at both tactile and kinesthetic levels. Besides, being able to manipulate objects, control and act on the virtual environment, is important to correctly feel immersed [58].

7.3 Solutions for Mitigation

Two solutions are currently investigated to mitigate the lateral motion failure. The first one consists in **using swarms of interfaces** (at least two [47, 60]) than can place themselves in a continuous manner for users to perceive the correct slippage (see Figure 12 - A). The second solution is to **move the interface as a function of the users motion direction** [34, 38] (see Figure 12 - B).

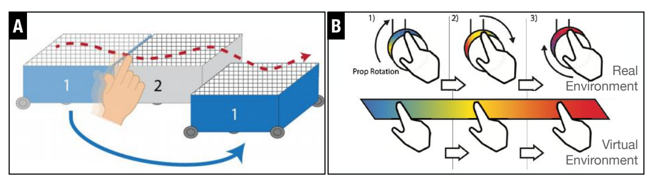

Figure 12: A. Swarms of Interfaces to display a continuous surface to explore [47]. B. Prop rotation enabling to perceive textures [38].

Similarly, as mobile interfaces struggle to display adequate perceived stiffness, they can for instance **move in the opposite direction** of the users' applied forces, to replicate the expected hardness. The normal force applied over the interaction contact area would therefore be increased and the hardness correctly simulated. On a side note, a force sensor could potentially be added over the contact surface to estimate when the contact will break, so the interface can anticipate when to stop this "opposite direction" movement.

Temperature-wise, apart from designing room-temperature experiences, we can integrate Peltier cells or heaters [45] to physically **replicate the adequate temperature** expectations. Suggestions regarding where the users should explore the objects (for instance by the handle of a hot teacup rather than by its base) can also be integrated **visually** in the virtual experience.

Finally, weight-wise, the easiest solution is to **exploit real objects or props** and to literally display them to the users. This removes the haptic transparency specification from the interface design, as the prop itself is being manipulated.

Regarding Structure-related properties, and in the same regards as with the Objects' Substance-related properties, the failures can potentially all be removed through the **use of real and untethered objects**. Yet, it is costly to have a multitude of objects to be displayed, especially in their entirety (to cater for the global and exact shape requirements). One solution could be to use objects of approximately the same sizes and primitives in the design of the VR scene [23], and to **rely on the users' vision to alleviate their haptic perception** (similarly as Section 6). Otherwise, **designing the VR scene accordingly with the available props** is a common technique: in the design of the on-demand handheld for instance, only spherical objects are available for interactions.

Mitigating the "weight" failures in the objects' structural properties can be achieved through the **integration of weight-shifting modules** within the ETHD. We can for instance imagine taking root from variable elastic stiffness [3], and attach the objects on the ETHDs with an elastic of variable stiffness. Thus, the same object could be perceived with different weights. We can potentially take root from *"property-changing proxies"* from [42] to integrate motors and actuators within the simulated objects, to relocate their centres of gravity and enable the user to perceive variable inertia using the same original prop.

8 Discussion & Future Work

We used the FMEA approach to identify failure modes, effects, and mitigation solutions of Encountered-type of Haptic Devices and provide an overview of ETHDs. We summarized the failure causes and their associated solutions in the following table (**Table 2**).

We note that some mitigation solutions are frequently used: for instance, adapting the virtual scene is the most obvious one. Reducing the number of objects to overlay or increasing the spacing between them enhances success in the Intentions, Displacement and Reconfiguration phases. We also note that many solutions take advantage of the visual dominance over haptics, to either redirect the user towards the physical objects; or simply use visual effects to distract the users from their current tasks (e.g. in Figure 5 - F, fireworks stop the user from reaching his target). A promising yet costly solution is also to increase the number of available robots within the physical arena: swarms robots can potentially overlay multiple objects of interest simultaneously, which benefit both of the designer and user experiences. Finally, the use of real objects as opposed to "re-configuring" into one seem as a reliable direction to provide the user with the correct haptic perception. We envision future work regarding both the FMEA approach itself and with ETHDs. We describe them in the subsequent subsections.

Failure Causes	Mitigation Solutions
Intentions - Step 1	
Algorithm delay	Scenario-based experience
Distance between OOI	Follow users
Number of OOI	Overlay closest object
Sizing of OOI	Decrease number of OOI
Lack of tracking information	Increase spacing between OOI
	Alter OOI sizes
	Increase number of robots
	Reduce users navigation speed
	Add prior probabilities
Displacement - Step 2	
Safety (Collisions)	Run simulations without users
Speed	Use avoidance algorithms
Accuracy	Distinguish user and ETHD workspaces
Precision	Add disturbing noise
Uncontrolled deceleration or oscillations	Use redirection techniques
Reachability	Use dynamic retargeting
	Add delay mechanisms
	Add color code
	Adapt virtual environment (e.g. # OOI etc)
Reconfiguration - Step 2.bis	
Shape Mismatch	Limit users tasks and objects
Resolution	Add <virtual:physical> mappings
Lack of physical coverage	Use Pseudo-Haptics
	Use Redirection techniques
	Match objects to device resolution
Interaction - Step 3	
Tactile slippage	Use swarms of robots
Perceived stiffness	Move in opposite direction
Lack of I/O capabilities	Use pseudo-haptics
Haptic transparency	Exploit real objects or props
Inertia	Adapt virtual environment
Edge touching	Integrate weight-shifting modules

Table 2: Potential Encountered-type of Haptic Devices Failures Causes and their Associated Mitigation Solutions, according to the ETHD scenario (Section 3.5).

8.1 FMEA & Constraint Functions

In this paper, we consider a wide range of ETHDs, i.e. we did not focus on a single prototype, and limited our analysis to the principal (sub-) functions. A thorough FMEA analysis can be achieved per-interface. This would consist in rating the failure modes, their severity and their probabilities, identifying the most critical failures and how to detect them and eventually to mitigate them for a given interface. This would then be summarized in an interface-dependent criticality matrix. We could also consider *constraint functions* such as their price in the designer conception, the noise or the clutter of the interface. For instance, drones (or even ungrounded mobile platforms) generate unwanted noise and wind that can alter user experience [2]. Two different ETHDs can also be compared, if they share the same application and therefore the same principal and constraint functions. As per [10], in which four ETHDs are compared, we can use the FMEA approach to identify common criteria across interfaces to enable their comparison. For instance, introducing the severity can also lead to understand what failures are the most critical: an ETHD first should be safe around users and provide in a second step the exact replica of its virtual counterpart.

8.2 Deployment & Validation

This FMEA approach helps identify the requirements and specifications of a given interface, therefore it can also help to identify how to refine them afterwards. This was used for a VR simulator [49] and can be adapted to other applications. It is efficient to highlight potential failures both from the designer and user perspectives. It still requires a long-term evaluation with experts to validate the usefulness and adoption of this approach on the field. Future work should investigate whether and how designers consider the different failures and their mitigation solutions in the design, implementation and evaluation of their systems. We expect that using this approach will facilitate deployments "beyond the prototype" [31].

9 Conclusion

In this paper, we provided an analysis of Encountered-type of Haptic Devices through their Failure modes, as a foundation for their development. This approach displays a groundwork for designers aiming to conceive ETHDs and emphasizes their main specifications.

We depicted a classic scenario involving such an interface, and identified the potential interfaces' failures modes, their associated effects and the solutions to mitigate them. We distinguished Designer *hardware* conception issues from User *interaction* perception failures. We then summarized the different failure causes and their associated solutions, and highlighted their similarities.

REFERENCES

[1] M. Abdullah, M. Kim, W. Hassan, Y. Kuroda, and S. Jeon. HapticDrone: An encountered-type kinesthetic haptic interface with controllable force feedback: Example of stiffness and weight rendering. In *2018 IEEE Haptics Symposium (HAPTICS)*, pp. 334–339, Mar. 2018. ISSN: 2324-7355. doi: 10.1109/HAPTICS.2018.8357197

[2] P. Abtahi, B. Landry, J. 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 - CHI '19*, pp. 1–13. ACM Press, Glasgow, Scotland Uk, 2019. doi: 10.1145/3290605.3300589

[3] M. Achibet, A. Girard, A. Talvas, M. Marchal, and A. Lecuyer. Elastic-Arm: Human-scale passive haptic feedback for augmenting interaction and perception in virtual environments. In *2015 IEEE Virtual Reality (VR)*, pp. 63–68. IEEE, Arles, Camargue, Provence, France, Mar. 2015. doi: 10.1109/VR.2015.7223325

[4] B. Araujo, R. Jota, V. Perumal, J. X. Yao, K. Singh, and D. Wigdor. Snake Charmer: Physically Enabling Virtual Objects. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction - TEI '16*, pp. 218–226. ACM Press, Eindhoven, Netherlands, 2016. doi: 10.1145/2839462.2839484

[5] ASQ. What is FMEA? Failure Mode & Effects Analysis | ASQ, 2021.

[6] Y. Ban, T. Kajinami, T. Narumi, T. Tanikawa, and M. Hirose. Modifying an identified curved surface shape using pseudo-haptic effect. In *2012 IEEE Haptics Symposium (HAPTICS)*, pp. 211–216, Mar. 2012. doi: 10.1109/HAPTIC.2012.6183793

[7] G. Binsted, R. Chua, W. Helsen, and D. Elliott. Eye–hand coordination in goal-directed aiming. *Human Movement Science*, 20(4-5):563–585, Nov. 2001. doi: 10.1016/S0167-9457(01)00068-9

[8] A. D. Blaga, M. Frutos-Pascual, C. Creed, and I. Williams. Too Hot to Handle: An Evaluation of the Effect of Thermal Visual Representation on User Grasping Interaction in Virtual Reality. p. 16, 2020.

[9] E. Bouzbib, G. Bailly, S. Haliyo, and P. Frey. CoVR: A Large-Scale Force-Feedback Robotic Interface for Non-Deterministic Scenarios in VR. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pp. 209–222. ACM, Virtual Event USA, Oct. 2020. doi: 10.1145/3379337.3415891

[10] E. Bouzbib, G. Bailly, S. Haliyo, and P. Frey. "Can I Touch This?": Survey of Virtual Reality Interactions via Haptic Solutions. In *32e Conférence Francophone sur l'Interaction Homme-Machine (IHM '20.21), April 13–16, 2021, Virtual Event, France*. Virtual Event, France, Apr. 2021.

[11] P. Bruns, C. J. Camargo, H. Campanella, J. Esteve, H. R. Dinse, and B. Röder. Tactile Acuity Charts: A Reliable Measure of Spatial Acuity. *PLoS ONE*, 9(2):e87384, Feb. 2014. doi: 10.1371/journal.pone.0087384

[12] L.-P. Cheng, T. Roumen, H. Rantzsch, S. Köhler, P. Schmidt, R. Kovacs, J. Jasper, J. Kemper, and P. Baudisch. TurkDeck: Physical Virtual Reality Based on People. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15*, pp. 417–426. ACM Press, Daegu, Kyungpook, Republic of Korea, 2015. doi: 10.1145/2807442.2807463

[13] A. Clarence, J. Knibbe, M. Cordeil, and M. Wybrow. Unscripted Retargeting: Reach Prediction for Haptic Retargeting in Virtual Reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 150–159. IEEE, Lisboa, Portugal, Mar. 2021. doi: 10.1109/VR50410.2021.00036

[14] B. Davies. Precision and accuracy in glacial geology.

[15] X. de Tinguy, T. Howard, C. Pacchierotti, M. Marchal, and A. Lécuyer. WeATaViX: WEarable Actuated TAngibles for VIrtual reality eXperiences. p. 9, 2020.

[16] D. Degraen, A. Zenner, and A. Krüger. Enhancing Texture Perception in Virtual Reality Using 3D-Printed Hair Structures. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pp. 1–12. ACM Press, Glasgow, Scotland Uk, 2019. doi: 10.1145/3290605.3300479

[17] L. Dominjon, J. Perret, and A. Lécuyer. Novel devices and interaction techniques for human-scale haptics. *The Visual Computer*, 23(4):257–266, Mar. 2007. doi: 10.1007/s00371-007-0100-4

[18] P. Fiorini and Z. Shiller. Motion Planning in Dynamic Environments Using Velocity Obstacles. *The International Journal of Robotics Research*, 17(7):760–772, July 1998. doi: 10.1177/027836499801700706

[19] E. J. Gonzalez, P. Abtahi, and S. Follmer. 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*, pp. 236–248. ACM, Virtual Event USA, Oct. 2020. doi: 10.1145/3379337.3415870

[20] F. Gonzalez. Contributions au développement d'une interface haptique à contacts intermittents. p. 162, 2015.

[21] V. Hayward and K. Maclean. Do it yourself haptics: part I. *IEEE Robotics & Automation Magazine*, 14(4):88–104, Dec. 2007. doi: 10.1109/M-RA.2007.907921

[22] Z. He, F. Zhu, and K. Perlin. PhyShare: Sharing Physical Interaction in Virtual Reality. *arXiv:1708.04139 [cs]*, Aug. 2017. arXiv: 1708.04139.

[23] 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 - CHI '16*, pp. 1957–1967. ACM Press, Santa Clara, California, USA, 2016. doi: 10.1145/2858036.2858134

[24] S. Hodges. Democratizing the Production of Interactive Hardware. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, pp. 5–6. ACM, Virtual Event USA, Oct. 2020. doi: 10.1145/3379337.3422877

[25] A. Horie, M. Y. Saraiji, Z. Kashino, and M. Inami. EncounteredLimbs: A Room-scale Encountered-type Haptic Presentation using Wearable Robotic Arms. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 260–269, Mar. 2021. ISSN: 2642-5254. doi: 10.1109/VR50410.2021.00048

[26] H. Hoshino. A Contruction MEthod of Virtual Haptic Space, 1995.

[27] H.-Y. Huang, C.-W. Ning, P.-Y. Wang, J.-H. Cheng, and L.-P. Cheng. Haptic-go-round: A Surrounding Platform for Encounter-type Haptics in Virtual Reality Experiences. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–10. ACM, Honolulu HI USA, Apr. 2020. doi: 10.1145/3313831.3376476

[28] H. Iwata. Locomotion Interfaces. In F. Steinicke, Y. Visell, J. Campos, and A. Lécuyer, eds., *Human Walking in Virtual Environments: Perception, Technology, and Applications*, pp. 199–219. Springer, New York, NY, 2013. doi: 10.1007/978-1-4419-8432-6_9

[29] K. B. Kaldestad, S. Haddadin, R. Belder, G. Hovland, and D. A. Anisi. Collision avoidance with potential fields based on parallel processing of 3D-point cloud data on the GPU. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3250–3257. IEEE, Hong Kong, China, May 2014. doi: 10.1109/ICRA.2014.6907326

[30] O. Khatib. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *The International Journal of Robotics Research*, 5(1):90–98, Mar. 1986. doi: 10.1177/027836498600500106

[31] R. Khurana and S. Hodges. Beyond the Prototype: Understanding the Challenge of Scaling Hardware Device Production. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–11. ACM, Honolulu HI USA, Apr. 2020. doi: 10.1145/3313831.3376761

[32] L. H. Kim and S. Follmer. Generating Legible and Glanceable Swarm Robot Motion through Trajectory, Collective Behavior, and Pre-attentive Processing Features. *ACM Transactions on Human-Robot Interaction*, 10(3):1–25, July 2021. doi: 10.1145/3442681

[33] Y. Kim, H. J. Kim, and Y. J. Kim. Encountered-type haptic display for large VR environment using per-plane reachability maps: Encountered-type Haptic Display for Large VR Environment. *Computer Animation and Virtual Worlds*, 29(3-4):e1814, May 2018. doi: 10.1002/cav.1814

[34] Y. Kim, S. Kim, U. Oh, and Y. J. Kim. Synthesizing the Roughness of Textured Surfaces for an Encountered-Type Haptic Display Using Spatiotemporal Encoding. *IEEE Transactions on Haptics*, 14(1):32–43, Jan. 2021. doi: 10.1109/TOH.2020.3004637

[35] R. Kovacs, E. Ofek, M. Gonzalez Franco, A. F. Siu, S. Marwecki, C. Holz, and M. Sinclair. Haptic PIVOT: On-Demand Handhelds in VR. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, UIST '20, pp. 1046–1059. Association for Computing Machinery, New York, NY, USA, Oct. 2020. doi: 10.1145/

3379337.3415854

[36] S. J. Lederman and R. L. Klatzky. Hand movements: A window into haptic object recognition. *Cognitive Psychology*, 19(3):342–368, July 1987. doi: 10.1016/0010-0285(87)90008-9

[37] W. A. McNeely. Robotic graphics: a new approach to force feedback for virtual reality. In *Proceedings of IEEE Virtual Reality Annual International Symposium*, pp. 336–341, Sept. 1993. doi: 10.1109/VRAIS.1993.380761

[38] V. Mercado, M. Marchal, and A. Lecuyer. ENTROPiA: Towards Infinite Surface Haptic Displays in Virtual Reality Using Encountered-Type Rotating Props. *IEEE Transactions on Visualization and Computer Graphics*, 27(3):2237–2243, Mar. 2021. doi: 10.1109/TVCG.2019.2963190

[39] V. Mercado and U. Rennes. Design and Evaluation of Interaction Techniques Dedicated to Integrate Encountered-Type Haptic Displays in Virtual Environments. p. 9, 2020.

[40] V. R. Mercado, T. Howard, H. Si-Mohammed, F. Argelaguet, and A. Lécuyer. Alfred: the Haptic Butler On-Demand Tangibles for Object Manipulation in Virtual Reality using an ETHD. In *2021 IEEE World Haptics Conference (WHC)*, pp. 373–378, July 2021. doi: 10.1109/WHC49131.2021.9517250

[41] V. R. Mercado, M. Marchal, and A. Lecuyer. "Haptics On-Demand": A Survey on Encountered-Type Haptic Displays. *IEEE Transactions on Haptics*, pp. 1–1, 2021. Conference Name: IEEE Transactions on Haptics. doi: 10.1109/TOH.2021.3061150

[42] N. C. Nilsson, A. Zenner, and A. L. Simeone. Propping up Virtual Reality with Haptic Proxies. *IEEE Computer Graphics and Applications*, p. 10, 2021.

[43] S. Razzaque, Z. Kohn, and M. C. Whitton. *EUROGRAPHICS 2001 / Jonathan C. Roberts Short Presentation © The Eurographics Association 2001. Redirected Walking*. 2001.

[44] A. Rodriguez-Fornells. Violating body movement semantics: Neural signatures of self-generated and external-generated errors. 2015.

[45] E. Shaw, T. Roper, T. Nilsson, G. Lawson, S. V. G. Cobb, and D. Miller. The Heat is On: Exploring User Behaviour in a Multisensory Virtual Environment for Fire Evacuation. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19*, pp. 1–13, 2019. arXiv: 1902.04573. doi: 10.1145/3290605.3300856

[46] K. Shigeta, Y. Sato, and Y. Yokokohji. Motion Planning of Encountered-type Haptic Device for Multiple Fingertips Based on Minimum Distance Point Information. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*, pp. 188–193. IEEE, Tsukaba, Mar. 2007. doi: 10.1109/WHC.2007.85

[47] A. F. Siu, E. J. Gonzalez, S. Yuan, J. B. Ginsberg, and S. Follmer. shapeShift: 2D Spatial Manipulation and Self-Actuation of Tabletop Shape Displays for Tangible and Haptic Interaction. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, pp. 1–13. ACM Press, Montreal QC, Canada, 2018. doi: 10.1145/3173574.3173865

[48] J. Snape, J. v. d. Berg, S. J. Guy, and D. Manocha. The Hybrid Reciprocal Velocity Obstacle. *IEEE Transactions on Robotics*, 27(4):696–706, Aug. 2011. doi: 10.1109/TRO.2011.2120810

[49] R. E. Sofronia, T. Knott, A. Davidescu, G. G. Savii, T. Kuhlen, and M. Gerressen. Failure mode and effects analysis in designing a virtual reality-based training simulator for bilateral sagittal split osteotomy: FMEA in designing a virtual reality-based training simulator for BSSO. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 9(1):e1–e9, Mar. 2013. doi: 10.1002/rcs.1483

[50] D. H. Stamatis. *Failure Mode and Effect Analysis: FMEA from Theory to Execution*. Quality Press, Jan. 2003. Google-Books-ID: TTxI8jbTkVwC.

[51] R. Suzuki, H. Hedayati, C. Zheng, J. Bohn, D. Szafir, E. Y.-L. Do, M. D. Gross, and D. Leithinger. RoomShift: Room-scale Dynamic Haptics for VR with Furniture-moving Swarm Robots. p. 11, 2020.

[52] R. Suzuki, E. Ofek, M. Sinclair, D. Leithinger, and M. Gonzalez-Franco. HapticBots: Distributed Encountered-type Haptics for VR with Multiple Shape-changing Mobile Robots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pp. 1269–1281. ACM, Virtual Event USA, Oct. 2021. doi: 10.1145/3472749.3474821

[53] N. Takizawa, H. Yano, H. Iwata, Y. Oshiro, and N. Ohkohchi. Encountered-Type Haptic Interface for Representation of Shape and Rigidity of 3D Virtual Objects. *IEEE Transactions on Haptics*, 10(4):500–510, Oct. 2017. doi: 10.1109/TOH.2017.2740934

[54] S.-Y. Teng, C.-L. Lin, C.-h. Chiang, T.-S. Kuo, L. Chan, D.-Y. Huang, and B.-Y. Chen. TilePoP: Tile-type Pop-up Prop for Virtual Reality. p. 11, 2019.

[55] J. van den Berg, Ming Lin, and D. Manocha. Reciprocal Velocity Obstacles for real-time multi-agent navigation. In *2008 IEEE International Conference on Robotics and Automation*, pp. 1928–1935. IEEE, Pasadena, CA, USA, May 2008. doi: 10.1109/ROBOT.2008.4543489

[56] 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, Los Angeles, CA, USA, 2017. doi: 10.1109/VR.2017.7892233

[57] A. Wexelblat. Virtual reality: applications and explorations, 1993. Myron Krueger, Artificial reality 2 An easy entry to Virtual reality Chap 7.

[58] B. G. Witmer and M. J. Singer. Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoperators and Virtual Environments*, 7(3):225–240, June 1998. doi: 10.1162/105474698565686

[59] 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 - SUI '16*, pp. 43–46. ACM Press, Tokyo, Japan, 2016. doi: 10.1145/2983310.2985746

[60] Y. Yixian, K. Takashima, A. Tang, T. Tanno, K. Fujita, and Y. Kitamura. ZoomWalls: Dynamic Walls that Simulate Haptic Infrastructure for Room-scale VR World. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*, UIST '20, pp. 223–235. Association for Computing Machinery, New York, NY, USA, Oct. 2020. doi: 10.1145/3379337.3415859

[61] Y. Yokokohji, R. Hollis, and T. Kanade. What you can see is what you can feel-development of a visual/haptic interface to virtual environment. In *Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium*, pp. 46–53. IEEE, Santa Clara, CA, USA, 1996. doi: 10.1109/VRAIS.1996.490509

[62] Y. Yokokohji, R. L. Hollis, and T. Kanade. WYSIWYF Display: A Visual/Haptic Interface to Virtual Environment. *Presence: Teleoperators and Virtual Environments*, 8(4):412–434, Aug. 1999. doi: 10.1162/105474699566314

[63] Y. Yokokohji, N. Muramori, Y. Sato, and T. Yoshikawa. *Haptic Display for Multiple Fingertip Contacts Based on the Observation of Human Grasping Behaviors*. 2005.

[64] 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 - CHI '18*, pp. 1–12. ACM Press, Montreal QC, Canada, 2018. doi: 10.1145/3173574.3174118