VoxelHap: A Toolkit for Constructing Proxies Providing Tactile and Kinesthetic Haptic Feedback in Virtual Reality

Martin Feick
martin.feick@dfki.de
DFKI and Saarland University, Saarland Informatics Campus
Saarbrücken, Germany

Cihan Biyikli
cihan.biyikli@dfki.de
DFKI and Saarland University, Saarland Informatics Campus
Saarbrücken, Germany

Kiran Gani
kiran.gani@dfki.de
DFKI and Saarland University, Saarland Informatics Campus
Saarbrücken, Germany

Anton Wittig
anton.wittig@dfki.de
DFKI and Saarland University, Saarland Informatics Campus
Saarbrücken, Germany

Anthony Tang
Singapore Management University
Singapore, Singapore
tonytang@utoronto.ca

Antonio Krüger
antonio.krueger@dfki.de
DFKI and Saarland University, Saarland Informatics Campus
Saarbrücken, Germany

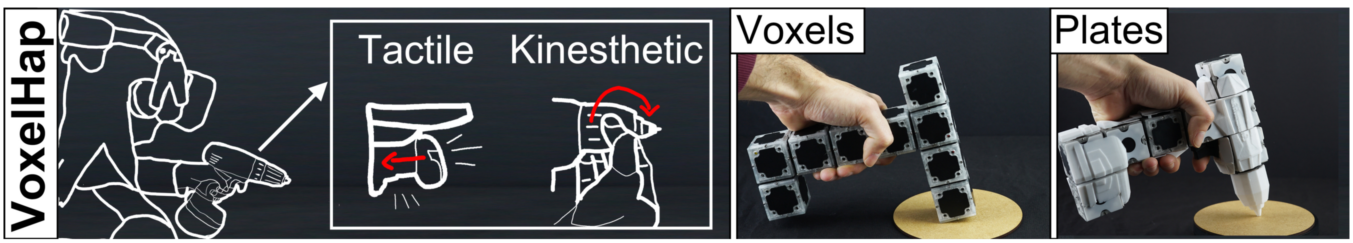

Figure 1: Lisa wants to use a power drill for training in VR, requiring a dedicated proxy that replicates form, feel and function.

ABSTRACT

Experiencing virtual environments is often limited to abstract interactions with objects. Physical proxies allow users to feel virtual objects, but are often inaccessible. We present the VoxelHap toolkit which enables users to construct highly functional proxy objects using Voxels and Plates. Voxels are blocks with special functionalities that form the core of each physical proxy. Plates increase a proxy's haptic resolution, such as its shape, texture or weight. Beyond providing physical capabilities to realize haptic sensations, VoxelHap utilizes VR illusion techniques to expand its haptic resolution. We evaluated the capabilities of the VoxelHap toolkit through the construction of a range of fully functional proxies across a variety of use cases and applications. In two experiments with 24 participants, we investigate a subset of the constructed proxies, studying how they compare to a traditional VR controller. First, we investigated VoxelHap's combined haptic feedback and second, the trade-offs of using ShapePlates. Our findings show that VoxelHap's proxies outperform traditional controllers and were favored by participants.

CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**.

KEYWORDS

Virtual Reality, haptics, toolkit, reconfigurable, tactile, kinesthetic

1 INTRODUCTION

Virtual Reality (VR) is a promising technology for many areas such as rehabilitation, simulation, training and design [25]. Although VR developers continue to enhance VR devices with increased visual and auditory capabilities, we still lack appropriate haptic feedback for VR objects; in fact, from a haptic perspective, VR content remains strictly virtual. One concept to address this shortcoming is the use of real-world 'stand-ins' for virtual objects, so-called *proxies*, that approximate virtual objects properties and characteristics [9, 27, 41, 50]. Ideally, one proxy can act as a stand-in for multiple virtual objects, reducing the number of proxies needed. To realize this, we need to overcome the challenge of *similarity*—how closely a proxy resembles a virtual object in terms of properties such as shape and size [3, 14], weight [63] or textures [12, 15]. In this work, we aim to enable users to construct highly functional proxies [14, 36], approximating the (for them) crucial properties of the virtual object to unlock realistic VR experiences.

Imagine Lisa, a carpeting trainee who wants to improve her practical skills without using valuable resources. Lisa can use the VR training program, but she would need dedicated controllers for each of the tools she wants to practice with. A single reconfigurable device could address this problem: it could provide realistic haptic feedback to be a hammer, or a power drill, because it can change its shape to adapt to the particular needs of the moment. This is where VoxelHap comes in, allowing Lisa to build a functional proxy drill using reconfigurable Voxels, approximating form (see Figure 1). As she wants to practice working with the drill, the proxy needs to support the desired functions: for example, provide tactile feedback when pressing the trigger button or kinesthetic haptic feedback when securing the drill bit. Lisa can add this functionality by using two special types of Voxels, Vibration and Rotation. Finally, she increases the shape fidelity of the proxy by 3D-printed ShapePlates with the aim to increase overall realism and usability.

In this work, we designed and developed VoxelHap: a block-based construction toolkit that enables users to build fully functional proxy objects that deliver tactile and kinesthetic haptic sensations. In its current implementation, VoxelHap combines and supports a range of haptic dimensions while enabling realistic input and output controls. Our main goal with VoxelHap proxies is to allow users to construct proxies with great expressibility, combining kinesthetic and tactile haptic feedback. To this end, we only consider handheld-sized objects. Here, we make five contributions:

1. We present the VoxelHap concept, empowering users to build functional proxies, providing tactile and kinesthetic feedback.
2. We designed and implemented VoxelHap's hardware, mainly consisting of Voxels and Plates.
3. We constructed several functional proxies using VoxelHap to evaluate the toolkit and demonstrate its versatility and power.
4. We conducted a technical evaluation and two user studies to better understand VoxelHap's capabilities and limitations.
5. We open-source the VoxelHap toolkit to the community[1].

2 RELATED WORK

Our work is positioned in the field of haptic feedback in VR and reconfigurable devices. Therefore, we use the next section to outline the most relevant work in the field.

2.1 Haptics in VR

The VR community broadly distinguishes between two types of haptic feedback, active and passive. The latter uses physical "stand-ins" (proxies) objects [27, 41] to approximate properties of virtual objects such as shape and size [3, 14], weight [63] or textures [12, 15]. Various studies have shown that proxies improve task performance, realism and thus presence in VR compared to purely virtual experiences—and even rough approximations have positive effects on the experience [24, 40]. In addition, researchers have demonstrated that small discrepancies between physical proxy and virtual object can remain unnoticed [17, 32, 58, 64]. This is caused by the visual dominance phenomenon, where, in the case of conflicting sensory modalities, vision usually dominates over other senses [21]. This effect has been utilized to increase the haptic resolution of passive proxies. For example, Ban et al. [5] manipulated the perceived shape of a proxy by offsetting a user's virtual hand from the real hand while maintaining its haptic cues. This method can also be used to increase the functional capabilities of proxies. For example, Yang et al. [61] presented a grabbing tool using only passive mechanisms and applied dynamic visual adjustments to create an illusion of different sized virtual objects. Similarly, Strandholt et al. [53] offset physical (proxy) tools, e.g., a hammer, to simulate a realistic haptic impact after each strike on a virtual nail. Such approaches are cheap and effective, but are often inflexible.

2.2 Vibrotactile Haptic Feedback in VR

In contrast, active haptic feedback uses computer-generated actuation to impose forces onto the human body, creating the desired sensations. There exist many approaches that use actuation to provide compelling haptic feedback for various types of interactions [2, 23, 44, 46, 54, 56, 59, 62]. Vibrotactile actuation is a form of active haptic feedback and has been extensively used in VR, ranging from simulating buttons [44] to object properties such as compliance [30], all the way to textures [54] and dynamic masses [56]. One downside is that such devices are often disembodied from the virtual content in terms of other object properties such as shape and size, restricting the possible interactions to a limited pre-defined set. Illusions might help to overcome this limitation, but rough approximations are still needed to create compelling proxies that enrich VR experiences [14, 40, 41].

2.3 Reconfigurable Devices for VR Haptics

Here, reconfigurable devices can help, because they can change their physical configuration, in a computer-controlled [51, 65] or manual way [14, 35, 45, 66], allowing them to adapt to the given use case.

Zhu et al. [66] used Rubik's twists, a passive low-cost twistable artifact, allowing users to build interactive haptic proxies for various hand-graspable VR objects. The shape approximations can be equipped with active components, e.g., buttons, to increase the range of supported interactions. On the other hand, Zhao et al. [65] presented block-based swarm robots that self-assemble physical handheld-sized proxies for VR, but are limited to block-like shapes. Therefore, *HapticBots* [55] extended this approach by outfitting the swarm robots with a shape-changing mechanism. Going beyond approximating shape and size, Li et al. [36] presented *HapLinkage*, allowing realistic simulations of e.g., wrenches, pliers, scissors and syringes by using a linkage mechanism that supports typical motion patterns. However, for novices it may be challenging to construct. Block-based approaches have been shown to allow lay users to easily build rough approximations [14, 31]. Therefore, Feick et al. [14] and Arora et al. [3] presented block-based construction toolkits, *TanGi* and *VirtualBricks*, for approximating size, shape but also functional parts. For example, rotational, stretchable and bendable parts of virtual objects. The authors highlight the benefits of block-based constructions with respect to their great modularity, scalability and ease-of-use.

In our work, we aim to empower users to replicate form, feel and function of virtual objects [14, 36, 66] without sacrificing reconfigurability. This is crucial, because visually and haptically realistically rendered objects [28, 42] promote better skill transfer into the real world. We utilize the advantages of passive haptics by using modular building blocks, called Voxels [43], for shape and size approximation, allowing the proxy to be touched and explored as a whole, going beyond pre-defined interaction points [13] as most VR controllers offer. To increase the haptic resolution of our proxies, we utilize rapid prototyping techniques [39], haptic illusions [1, 7, 30, 44] and active haptic components [2, 46, 54, 56].

3 VOXELHAP

VoxelHap is a toolkit that gives users the ability to create haptic proxies based on a set of building blocks. These building blocks give

[1] https://github.com/MartinFk/VoxelHap

Figure 2: BaseVoxel consists of a PCB in a block design, a translucent 3D-printed shell assembled with M3 screws, and conductive touch pads on each side.

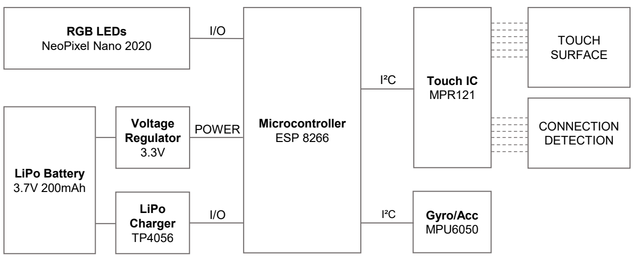

Figure 3: Schematics of a BaseVoxel.

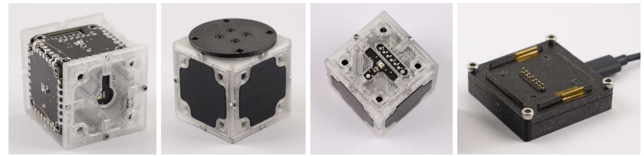

Figure 4: Displaying how the PCB fits inside the 3D-printed shell. The ConnectionPlate adapter of the RotationVoxel. The pin layout, LED and touch sensing pin, and right next to it the charging and flashing unit with pogo pins (left to right).

the proxies different physical and interaction capabilities, and employ haptic illusions to enhance the haptic resolution of the proxy. VoxelHap is an end-to-end system, where the software tool supports the assembly process by providing visual guidance to efficiently assemble the desired proxy. The system hosts an exact representation of each proxy in its current configuration in real time. At VoxelHap's core we have BaseVoxels, the most basic reusable unit. To increase haptic resolution, Voxels and Plates with special functionalities and properties can be added on demand. Below, we discuss the design rationale and fabrication of VoxelHap's components.

3.1 Voxels

We use cube blocks (Voxels) as our basic building structure [14, 31, 43, 45, 48, 49, 60], because cubes are a well-established approach for construction kits, and have been demonstrated to be effective. Voxels can be connected to each other genderlessly using a custom-designed ConnectionPlate mechanism (i.e., they can be connected in any direction, and in any orientation). Each Voxel is a self-contained unit and designed for input and output [60]. In this work, we present two additional types of Voxels that provide additional (corresponding) capabilties, VibrationVoxel and RotationVoxel. The toolkit can be extended to include new capabilities, e.g. twistable or bendable units, or other types of sensors [3, 14, 37, 60]. Voxels are rechargeable and flashable through a custom docking station with pogo pin connections (see Figure 2 and Figure 4).

BaseVoxel. The BaseVoxel offers the basis for all other types of Voxels. It measures 37 × 37 mm (outer shell) and weighs 38 g. Each BaseVoxel consists of six 3D FDM printed translucent PETG shells of 3mm thickness secured with M3 screws, and has touch sensitive areas on all sides, which have been 3D-printed with conductive filament. It hosts an Espressif ESP8266 (ESP12F Module), a TP4056 LiPo Charger IC, an MPU6050 six-axis gyro and accelerometer, a voltage regulator (LDL212D33R), a proximity capacitive touch sensor controller (MPR121), six NeoPixel Nano 2020 LEDs to illuminate the sides, and a 3.7V 200mAh LiPo for powering the Voxel. The PCB comes in a cube shape (33 × 33 mm) with a cut-out part for the antenna to ensure a proper fit inside the 3D-printed Voxel; the high level schematics can be found in Figure 3.

VibrationVoxel. This type of Voxel enables rendering of tactile feedback when holding or touching the proxy. We implemented this by extending the BaseVoxel by a voice-coil actuator (HAPTIC™ Reactor ALPSALPINE AFT14), a Bluetooth 5.0 low latency (45ms) audio receiver and an audio class D amplifier (PAM8403). Our PCB design ensures that the actuator can be placed and affixed inside the PCB cube. For ease of use, the vibration patterns (audio files) are sent directly from the PC via Bluetooth, i.e., once the VibrationVoxel is running it can be used as any other audio output device. It weighs 46 g.

RotationVoxel. Enables proxies to have rotational parts [3, 14, 36], but also renders kinesthetic force feedback for rotational movements. To achieve this, we added a brushless DC Motor2204 260KV, an ICTMC6300 motor driver and magnetic position sensor (MAQ473) to the BaseVoxel's PCB. The RotationVoxel can be used passively e.g., for sensing, or as an active component to provide resistances when turning it. To enable a solid connection, one side of the BaseVoxel and the PCB was removed to offer space for the ConnectionPlate adapter (see Figure 4). Total weight is 65 g.

Voxels enable rapid prototyping of proxies within several minutes [14], creating functional low fidelity versions of virtual models that may be sufficient in some situations, for example, to better understand depth and scale of virtual scenes [40].

3.2 Plates

Plates are optional passive components that increase haptic resolution and functionality. These allow the user to create higher fidelity haptic proxies, which can be used for realistic training and simulations. We took inspiration from Mueller et al. [39]'s work on *FaBrickation*, because similar to VoxelHap, the authors use basic block structures (LEGO®) to improve fabrication and prototyping time and additionally, they 3D-printed high fidelity parts for accurate shape approximation. Within VoxelHap, we currently provide five different types of Plates as depicted in Figure 5 and Figure 6: Connection-, Shape-, Texture-, Weight- and TrackingPlates.

ConnectionPlates. Required to connect Voxels to each other. They are 3D-printed using conductive filament, creating a closed circuit upon connection, allowing the system to determine the side and

Figure 5: 3D-printed ShapePlates of the drill that can be attached to the low fidelity drill to increase shape resolution.

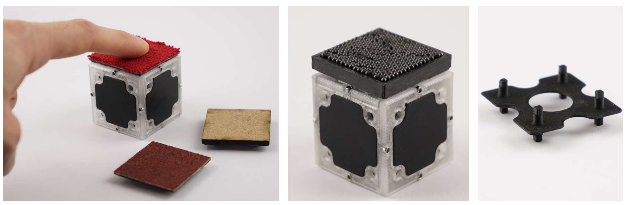

Figure 6: TexturePlates, WeightPlate and ConnectionPlate.

the type of Voxels that got connected to each other (see Figure 6).
ShapePlates. Can be used to increase proxy fidelity if needed (see Figure 5). For instance, any shape can be 3D-printed onto the Plates to increase shape resolution. The ShapePlates can be generated by first applying voxelization and second, projecting anything below a 45-degree angle onto the ShapePlate, because that is the max. overhang that can be reliability 3D-printed without a support structure. If 3D-printed with conductive filament, we can also detect touch input on ShapePlates, since they make contact with Voxel's touch pads, e.g., the power drill's trigger button shown in Figure 1.
TexturePlates. Plates that are augmented with material textures such as fabric or rubber. They can be used to simulate local differences in textures to e.g., improve the grip when holding a tool or, depending on the use case, create various touch sensations (see Figure 6 left). TexturePlates can also be combined with other fabrication techniques, such as 3D-printed hair structures by Degraen et al. [12] or metamaterials [15], to create different material texture sensations—the advantage being that it can be 3D-printed all together.
WeightPlates. The overall weight and center of mass are important properties of proxies [63] and hence, VoxelHap offers Plates to 'balance' them. To do so, we embedded lead into the Plates, allowing users to increase the overall weight. Moreover, the location of the WeightPlate will shift a proxy's center of mass. A WeightPlate adds 45 g to the proxy (see Figure 6 middle).
TrackingPlates. Voxels sense their orientation and acceleration, but for accurate mapping between real-world proxy and virtual model, a shared coordinate system is required. Here, TrackingPlates offer mounts for systems such as an HTC Vive tracker or Optitrack marker to enable robust tracking.

Overall, we fabricated 13 Voxels (9 BaseVoxels, 3 VibrationVoxels and 1 RotationVoxel) and several Plates of each type.

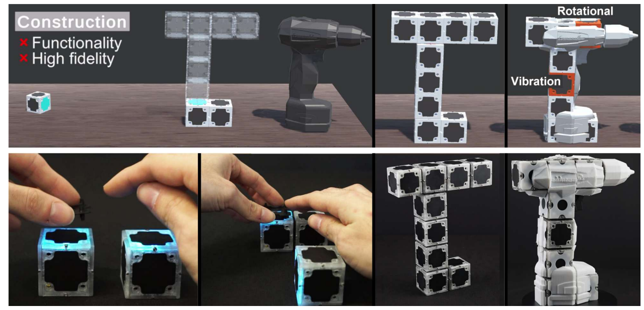

Figure 7: VoxelHap's supported construction process.

3.3 Haptic Proxy Description Format

While haptic proxies can be constructed haphazardly, we need a structured format to describe the appearance, behavior and functionality of block-like proxies. We introduce the Haptic Proxy Description Format (*.hpdf*), a semi-generalizable description format for proxy design, which includes a proxy's kinematics, a functionality log and a construction plan. To this end, users need to manually create the *.hpdf* once and thence, they can re-use or share it.
Kinematics. 3D models for different parts of the object as .fbx files, especially for functional parts. This includes physical properties of object parts such as their texture, material and mass, and also mechanical features to allow realistic approximation.
Functionality Log. Contains a list of parametric input and output controls such as button press and the resulting visual and haptic output.
Construction Plan. A visual construction plan where a user can decide between three levels of proxy fidelity: (1) a low fidelity approximation only requiring BaseVoxels, (2) high shape fidelity by adding ShapePlates, and (3) functionality through the use of Vibration- and/or RotationVoxels.

3.4 Construction Walkthrough

In the following, we illustrate the construction process (Figure 7), where Lisa downloads the *.hpdf* file for the desired power drill. As soon as she powers the Voxels, they connect wirelessly, and appear in the virtual environment. As she engages with the assembly process, two Voxels light up to indicate which sides should be connected. As Lisa follows the step-by-step instructions, different sides light up until she is finished with the low fidelity proxy. She wants to increase the shape resolution and therefore, she selects the ShapePlate option. As a result, she needs to 3D-print the missing pieces to achieve the desired shape resolution. Similarly, she can select functional features for the proxy. Lisa opts for the trigger button and three haptic renderings: vibrotactile feedback when using the trigger button, as well as for the drill running in idle and drilling mode. For the latter, she can map interactions with virtual objects to their corresponding haptic feedback pattern; for instance, the haptic feedback when drilling wood is different from that of steel or concrete. After Lisa has practiced using her power drill proxy, she wants to try a different type of drill, e.g. with two handles to improve stability. Thus, she simply goes through the steps outlined before, reconfiguring her proxy.

Figure 8: VoxelHap rattle. Low shape fidelity, high shape fidelity and corresponding virtual model (left to right).

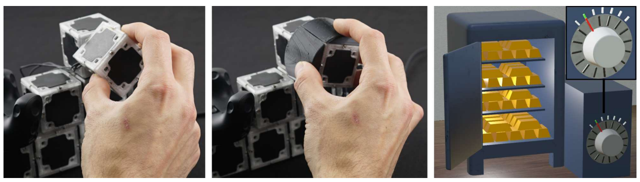

Figure 9: VoxelHap dial lock. Low shape fidelity, high shape fidelity and corresponding virtual model (left to right).

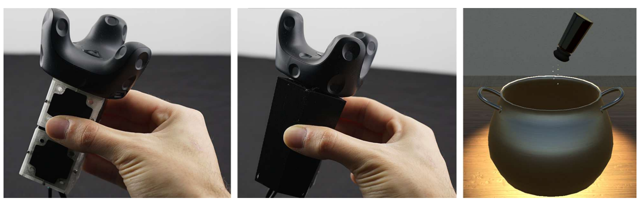

Figure 10: VoxelHap salt shaker. Low shape fidelity, high shape fidelity and corresponding virtual model (left to right).

4 CONSTRUCTING VOXELHAP PROXIES

In this section, we showcase VoxelHap proxies' expressiveness power across three dimensions: types of functionality, haptic feedback and geometry. We did this as a first evaluation of the toolkit as proposed by Ledo et al. [34].

4.1 Types of Functionalities

One of the VoxelHap toolkit's main contributions are replicas that combine tactile and kinesthetic haptic feedback. To illustrate this, we built two functional proxies, a rattle and a coded dial lock.

Rattle. We chose a matraca toy (see Figure 13), because it nicely demonstrates tactile and kinesthetic feedback within a single interaction. When rattling, a user can feel the moving parts (inertia) and the distinct clacks of the cogwheel. We assembled this proxy using four BaseVoxels, a VibrationVoxel and the RotationVoxel. The BaseVoxels allow a rough approximation of the rattle, but more importantly the RotationVoxel enables the rotating parts and thus renders the kinesthetic feedback (inertia). Tactile sensations of the cogwheel are then added at each cog (35°) using the VibrationVoxel. By 3D-printing ShapePlates, the handle becomes more realistic and ergonomic (see Figure 8).

Coded Dial Lock. A coded dial lock is another example where tactile feedback (tick marks) and kinesthetic feedback (resistance) is needed to provide compelling haptic feedback. The VoxelHap proxy consists of six BaseVoxels to create a supporting structure, approximating the height of a stationary dial lock, a VibrationVoxel to haptically render the tick marks, and the RotationVoxel, for sensing and for providing basic resistance when turning (see Figure 9). This is an example of an ungrounded haptic devices.

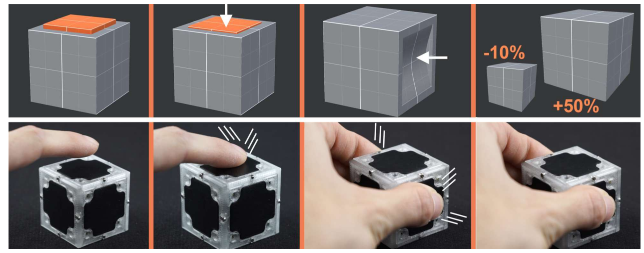

Figure 11: Pseudo-haptic button, stiffness and size variations.

4.2 Types of Haptic Feedback

Using the VibrationVoxel and RotationVoxel with the touch/force sensing capabilities gives us the ability to create a variety of highly synchronized haptic impressions and illusions. Here, we present a set of functional proxies that we constructed using VoxelHap.

Pseudo-Haptic Illusions. Combining pseudo-haptic effects [20, 33] with vibrotactile actuation [23] can result in realistic haptic sensations. For example, Park et al. [44] simulated a physical button press. As shown in Figure 11, we re-implemented this using VoxelHap, coupling touch sensor input (pressure) to the corresponding pseudo-haptic visualization and tactile feedback. Similarly, we can alter the perceived stiffness of the VibrationVoxel when pinching it. This can be achieved by mapping the force input to a mesh deformer[2] and the corresponding vibrations analogously to [1, 8, 30]. For this type of interaction, we can also alter the size of the virtual model while using the same proxy. Bergström et al. [7] showed that virtual objects can be up to 50% larger or 10% smaller compared to their physical counterpart. All implemented illusions are depicted in Figure 11.

Rendering Textures. Interacting with surface textures results in unique haptic feedback. For instance, the feeling when drilling wood differs significantly from that of drilling metal. Our VoxelHap power drill proxy (see Figure 1) allows users to perceive this difference through varying vibrotactile feedback [54]. Finally, TexturePlates can be used to provide tactile feedback when touching proxies in an encounter-type fashion [2, 54], i.e., rotating to the corresponding textures when needed.

Simulating Mass. We can simulate coffee sloshing around in a mug, e.g., through vibrotactile feedback [56]. To demonstrate this, we built a mug, consisting of a VibrationVoxel, a BaseVoxel and ShapePlates (see Figure 12). Even more realistic sensations can be achieved by adding the RotationVoxel and a WeightPlate. Synchronizing the moving WeightPlate with the interaction creates a realistic feeling of inertia [46, 59]. This effect may be strengthened when combined with pseudo-haptic weight techniques [47].

[2]https://assetstore.unity.com/packages/tools/modeling/deform-148425

Figure 12: VoxelHap mug. BaseVoxel, VibrationVoxel and a mug handle using ShapePlates. Full shape resolution using ShapePlates. Added RotationVoxel and WeightPlate for moving mass simulation (left to right).

4.3 Types of Geometries

VoxelHap's core capability is limited by the Voxel's size. To increase shape resolution, one may add ShapePlates; however, this requires additional fabrication time. Moreover, the block-based approach favors certain types of geometries.

Our current implementation benefits convex object geometries, but only for objects bigger than the BaseVoxel size of 3.7 *cm*. Any arbitrary shape can be 3D-printed onto the ShapePlates to resemble smaller features of the virtual object, for instance, a trigger button, handles, or even other geometric primitives e.g., spheres, triangles or pyramids. One limitation is that the construction of high-fidelity objects such as pens or screwdrivers [36] may be 3D-printed onto a ShapePlate, but interactions will often be limited to touch and exploration, rather than function. On the other hand, concave geometries can also be achieved. For instance, a ring may be built with BaseVoxels and ShapePlates, but this may quickly become bulky, depending on the use case.

5 IMPLEMENTATION

5.1 Voxel Firmware

The Voxel software is implemented in C++, and uses an ESP-WIFI MESH[3] to communicate with a master (ESP32), which is tethered to a host machine using serial port communication. By using ESP-MESH, we ensure low power consumption, low latency and great scalability, theoretically up to 1000 Voxels. There exist two template commands, for sensing and actuation, which are used to enable seamless communication between Voxels and the master. Our master application offers an exact representation of the current types of Voxels used, their functionality and status. Moreover, we can support and monitor multiple proxies at the same time. Once a users powers on a Voxel of any type, it automatically registers at the master with a unique id, and calibrates itself. Voxels can detect their neighbors when being connected to each other using conductive pins on the ConnectionPlates, and pads on the underlying PCB. They constantly send updates to the master (baud rate 115200) to enable seamless interactions. The RotationVoxel was controlled using SimpleFOC [52].

[3]https://www.espressif.com/en/products/sdks/esp-wifi-mesh/overview

Figure 13: Recording vibrations from real-world object interactions using a condenser vibration pickup microphone.

5.2 Virtual Environment

Our Voxel pipeline works with any machine and software that supports serial port communication. We implemented our virtual environment using Unity3D (v.2021.3.10f1). Sensing updates are received asynchronously and can be mapped to the desired visualization. Actuation commands can be triggered within the virtual environment, and are then forwarded to the corresponding Voxel. To this end, we provide a basic set of functions as illustrated before; however, additional functionality can be added as needed.

Construction Plan. Voxelizing the virtual model was done using the mesh voxelizer asset[4]. Currently ShapePlates have to be manually post-processed in CAD software. When initiating the construction process, the system sends two actuation commands (i.e., turn LED on) to a pair of Voxels. This lights up one side on each Voxel, indicating which sides should be connected [31]. Connecting to the wrong side will prevent users from continuing. Once either Voxel has established a connection, they send a confirmation event. The construction plan retrieves information from the manually generated *.hpdf* file.

5.3 Recording Vibrotactile Feedback

Vibrotactile feedback can offer rich haptic feedback, but is challenging to program [11]. Inspired by Tanaka et al. [56], we attached an AKG C411 high-performance miniature condenser vibration pickup microphone to the real-world objects, recording the vibrations caused by the interaction on a Zoom H4n Pro, as depicted in Figure 13. Through early pilot testing, we found that simply replaying the recorded sounds on the actuator did not result in convincing haptic feedback. The sensations felt rather weak and pilots struggled to distinguish them. Therefore, we modulated the recorded sound using a square shape wave with an amplitude corresponding to the recorded volume (in *dB*). We synchronized the vibrotactile feedback using discrete keypoints in the interaction.

6 TECHNICAL EVALUATION

We evaluated the technical capabilities of VoxelHap with respect to the following aspects. The table below summarizes the results: (1) according to ESP-MESH up to 1000 Voxels; however, we can only guarantee 13 Voxels; (2) measured weight of the proxies (excluding trackers), except dial lock, because it was stationary; (3) measurements were taken using a PCE-FM 50N Series force gauge and we report average values after 3 repetitions. (4) and (5) are hardware limitations; (6) fabrication costs including all hardware components.

[4]https://assetstore.unity.com/packages/tools/utilities/mesh-voxelizer-150233

1	Maximum # of Voxels	13
2.a	Shaker Weight	85 g
2.b	Rattle Weight	339 g
2.c	Shaker Weight With ShapePlates	105 g
2.d	Rattle Weight With ShapePlates	368 g
3.b	ConnectionPlate Strength Vertically	36.56 Nm
3.c	ConnectionPlate Strength Horizontally	7.87 Nm
3.d	ConnectionPlate Req. Connection Force	1.26 Nm
3.e	RotationVoxel Rotational Torque Range	0.3–17.2 Nmm
4	RotationVoxel Rotational Range	∞
5	RotationVoxel Max. Rotational Velocity	260 RPM
6.a	BaseVoxel Fabrication Cost	€19.87
6.b	RotationVoxel Fabrication Cost	€58.28
6.c	VibrationVoxel Fabrication Cost	€37.99
7.a	BaseVoxel & RotationVoxel Est. Latency	48 ms
7.b	VibrationVoxel Est. Latency	56 ms
8.a	BaseVoxel Avg. Battery Power	82 min
8.b	RotationVoxel Avg. Battery Power	26 min
8.c	VibrationVoxel Avg. Battery Power	63 min

Table 1: Summary of technical evaluation.

(7) was estimated by combining latency of the different components.
(8) is runtime under 50% load when fully charged.

7 USER EVALUATION

To further evaluate the toolkit, we conducted two user studies, comparing VoxelHap against the current state of the art [34]. First, we wanted to understand potential benefits of coupling function with the corresponding tactile and kinesthetic haptic feedback provided by VoxelHap. Second, we studied VoxelHap's high shape approximation feature ShapePlates.

7.1 Experiment 1—The Impact of Functionality

The first study was designed to compare low fidelity BaseVoxel approximations with combined functional haptic feedback against the current state of the art, a standard Vive controller providing basic vibrotactile feedback. We allowed participants to customize the strength of the vibrotactile feedback, for both devices, until it felt realistic to them. We included this to collect more insights, potentially informing future research on designing vibrotactile feedback for VR. Here, we only use Base and Functional Voxels. Please note, the Vive controller uses a rumble motor instead of a voice-coil actuator. Therefore, we could not play the recorded sounds. The duration of the vibrations was tailored to fit the interaction, and strength was adjusted by changing the vibration's amplitude.

7.1.1 Task & Proxies. We implemented three mini-games, highlighting VoxelHap's functional capabilities. We used the Rattle, Dial Lock and Shaker proxies shown above. Since we only fabricated a single RotationVoxel, we built a second stationary device with the RotationVoxel's hardware, except the PCB. This was used for the safe dial lock. In the following, we describe the three mini-games.

Dial Lock. The goal of this task to unlock the safe. To do so, participants had to rotate the dial to different positions highlighted in red. They had to remain in that position for 500ms before the next target appeared. Six combinations had to be solved to unlock the safe. The order was randomized; however, the total rotational travel distance required to complete the task remained equal.

Rattle. Here, participants were asked to cheer for their favourite team by rattling with a specific rotational speed in order to fill the progress bar. The progress bar filled quicker if they stayed within the correct speed range. This was done to limit participants' maximum velocity while still allowing them to focus on the haptic feedback provided.

Shaker. Participants were asked to shake salt into the highlighted pots. The amount of salt required to complete one pot was set to three successful shakes, i.e., at least $\frac{3}{4}$ of the salt needed to land in the pot. To finish the game, six completed pots were required, resulting in $3 \times 6 = 18$ interactions. The order of the highlighted pots was randomized.

We opted for the three VoxelHap proxies, because they combine tactile and kinesthetic haptic feedback, differ in dimensions, and cover two types of devices, grounded as well as ungrounded. We decided against proxies that utilize VR illusions, because they need to be carefully calibrated to an individual is perceptual boundaries in order to remain unnoticed [19]. Otherwise, they might affect participants' assessments.

7.1.2 Design. We used a within-subjects design. We had three conditions: Dial Lock, Rattle and Shaker, each performed once with a VoxelHap proxy and the standard Vive controller. The conditions were counterbalanced using a Latin square and we alternated the order of VoxelHap/controller with each participant. This way, we ensured that each participant was able to directly compare devices.

7.1.3 Participants. We recruited 12 right-handed participants (four female, eight male), aged 18–27 (M : 24.15; SD : 3.02) from the general public and the local university. Participants had a range of different educational and professional backgrounds including computer science, linguistics and data science. All participants reported normal or corrected-to-normal vision and did not report any known health issues which might impair their perception. Three participants had never used VR before, six had used it a few times (one to five times a year), two reported using it often (6–10 times a year), and one participant used it on a regular basis (more than 10 times a year). Participants not associated with our institution received €10 as remuneration for taking part in the experiment. The study was approved by the University's Ethics Board.

7.1.4 Apparatus. We used an apparatus consisting of an HTC VIVE Pro Eye tracking system with SteamVR (v.1.22) and OpenVR SDK (v.1.16.8). The simple virtual scene was developed in Unity3D (v.2021.3.10f1). We used an Acer Predator Orion 5000 PO5-615s offering an Intel® Core i9 10900k CPU, 32 GB RAM and an Nvidia® GeForce RTX 3080 for running the experiment. VoxelHap proxies were tracked using Vive trackers (v.3). In order to avoid potential issues with battery power, we connected VoxelHap proxies to a power source for the study.

7.1.5 Procedure. Participants were given a general introduction to the study. Following this, we obtained consent, and asked them to fill in a demographics questionnaire. Next, we showed and explained

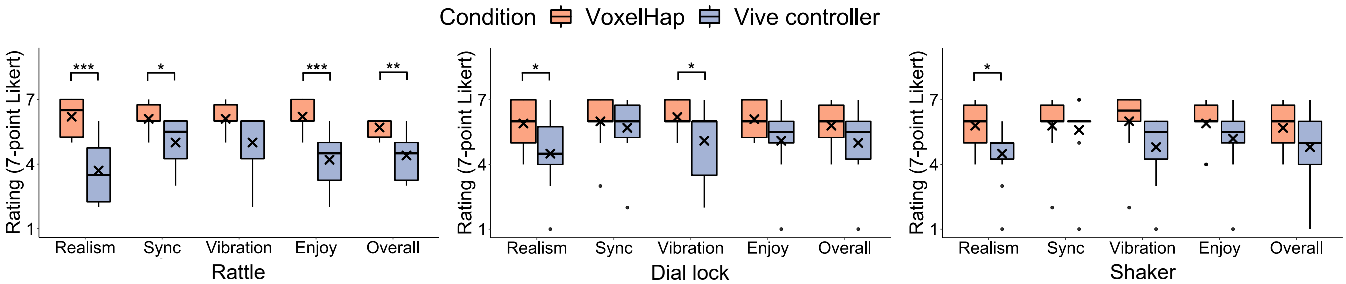

Figure 14: Experiment 1 results. VoxelHap proxies against Vive controller. * = p < .05, ** = p < .01, *** = p < .001.

the VoxelHap proxies to the participants to ensure that they were familiar with the devices. Then, they entered the VR environment and were guided through a practice round before they started with the main task. During the warm-up round they were asked to adjust the strength of the vibrotactile feedback for both devices until it felt realistic to them. Participants were instructed to perform a mini-game with each of the devices. After they completed the mini-game a questionnaire appeared, assessing their experience with the device. After each condition (i.e. they finished a mini-game with both devices) participants were asked to name their favorite and the most realistic device and, if possible, explain why they preferred it over the other. The total experiment took about 45 min.

7.1.6 Data Collection. We collected data from five sources: a pre-study questionnaire for demographic information; field notes and observations; configured strength of haptic feedback; a questionnaire after each task and condition in VR using the VRQuestionnaireToolkit [16] and a semi-structured interview to better understand participants' experiences with the system.

The questionnaire items were adapted from prior work [14, 63]:
(1.1) My interactions felt realistic.
(1.2) The vibrations were in sync with my interactions.
(1.3) The vibrations matched my visual impression of the game.
(1.4) I enjoyed playing the game.
(1.5) Overall impression of the game.
(2.1) Which of the two devices did you enjoy more?
(2.2) Which of the two devices felt more realistic?

7.1.7 Analysis. Statistical tests were chosen based on whether the data satisfied parametric test assumptions at $\alpha = .05$ using Shapiro–Wilk tests and QQ-plots. For outlier removal, we used the box plot method. We corrected pairwise comparisons using Bonferroni-Holm adjustments. Semi-structured interviews were coded for qualitative analysis.

7.1.8 Results. Overall, our three VoxelHap proxies, Dial Lock, Rattle and Shaker received better scores compared to the standard Vive controller across all five questions regarding realism, synchronization, haptic feedback, enjoyment and overall impression (see Figure 14). Differences were most clear for the Rattle, which saw the most significant differences, followed by the Dial Lock and the Shaker. Participants reported significantly higher levels of realism and enjoyment with the VoxelHap Rattle. This is supported by our analysis on which device participants associated with the highest realism. As depicted in Figure 14, participants overwhelmingly answered VoxelHap proxies. When asking participants which device they enjoyed the most, the VoxelHap Rattle and Dial Lock received comparable results, despite the Shaker being on par (see Figure 17). Here, some participants stated that the ergonomics of the Vive controller and the basic haptic feedback were "good enough" (P8) to play the game. Further, they expected the Dial Lock to have "more friction" (P6), which we intentionally kept low. Finally, participants chose VoxelHap proxies to feel more realistic: for example, the VoxelHap Rattle, due its "realistic and smooth motion" (P1) and the "weight balance [feeling] so nice" (P5), because "you can really feel the weight moving" (P7). The VoxelHap Shaker was selected because "shape and vibration were more realistic" (P6) and had a "similar shape" (P3) than a real salt shaker. Finally, participants preferred the VoxelHap Dial Lock, because it "was more intuitvie" (P3) and "you could actually feel it rotating" (P11).

Our study demonstrates that combining low fidelity approximation and function significantly increased realism in VR, and that VoxelHap is an effective toolkit to achieve this.

Vibrotactile feedback. We asked participants to configure the strength of the vibrotactile feedback during the warm-up phase in 10% increments (0%–unnoticeable; 100%–strongest vibration possible). They were instructed to select a strength that felt realistic to them. The results for both Vive controller, $r(34) = -.60, p < .001$, and VoxelHap, $r(34) = -.49, p = .002$, suggest a strong negative correlation between object and configured vibration strength, meaning that for the Shaker participants selected weaker vibrations than for the Dial Lock and the Rattle, respectively (see Figure 18). This is in line with our assumption that the three study objects differ in their "expected" feedback. Together with the findings above, receiving high ratings in synchronization and greater average scores for the provided vibrotactile feedback, this demonstrates that VoxelHap proxies effectively produce realistic haptic sensations.

7.1.9 Summary & Discussion. Our study showed that VoxelHap proxies outperform the current standard VR controllers. This shows VoxelHap's concept and implementation is effective and robust. However, we frequently observed uncanny valley of haptics effects [6]—where small differences affected participants' assessments, e.g., the low friction of the Dial Lock. Even though participants preferred the direct mapping between VoxelHap proxies, the virtual model and their interactions, they often highlighted the Vive controller's better ergonomics. This is not surprising, because block-like structures can be difficult to hold (e.g., the handle of the rattle). With

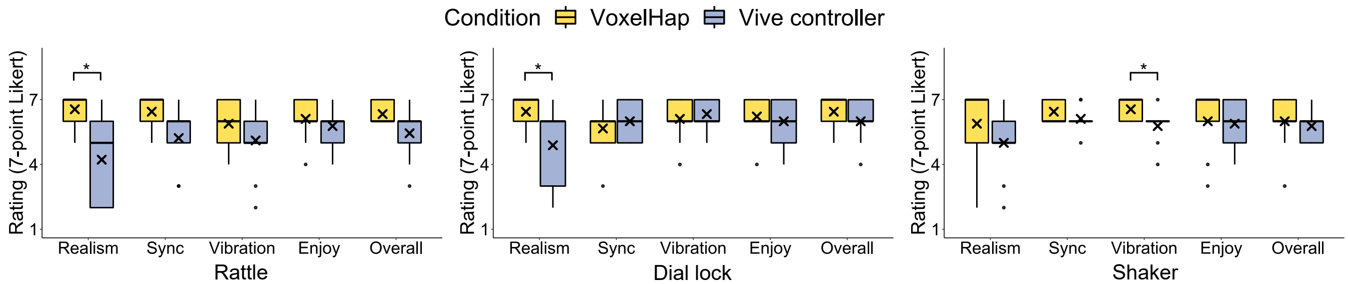

Figure 15: Experiment 2 results. VoxelHap proxies with ShapePlates against Vive controller. * = p < .05.

this in mind, we designed a second experiment, hypothesizing that adding VoxelHap's ShapePlates significantly improves user experience.

7.2 Experiment 2—The Impact of ShapePlates

Informed by the results of experiment 1, we conducted a second experiment. Our central interest lay in whether adding shape resolution through ShapePlates improves the experience; especially, how it influences realism, tactile feedback and synchronization that may be affected by adding an "additional" 3D-printed layer to the proxy. By doing so, we could study if our 3 Voxeltypes and ShapePlates work effectively together. We kept Task, Design, Procedure, Data Collection and Analysis the same. For the Shaker this meant that the proxy's size exceeded the virtual models size by more than recommended [7]. We included this (1) to study tradeoffs of ShapePlates and (2) to better understand the practical effects of VoxelHap's geometric limitations. Finally, we also asked participants to customize the resistance of the Dial Lock to be as realistic as possible.

7.2.1 Participants. We recruited a new set of 12 right-handed participants (five female, seven male), aged 19–32 ($M: 23.92; SD: 3.73$) from the general public and the local university. Participants had a range of different educational and professional backgrounds including psychology, computer science, law, cybersecurity, economics, biology and visual computing. All participants reported normal or corrected-to-normal vision and did not report any known health issues which might impair their perception. Four participants had never used VR before, six had used it a few times (one to five times a year), one reported using it often (6–10 times a year), and another participant used it on a regular basis (more than 10 times a year). Participants not associated with our institution received €10 as remuneration for taking part in the experiment. The study was approved by the University's Ethics Board.

7.2.2 Results. Our results confirm the findings from experiment 1. VoxelHap proxies received greater average scores than the Vive controller. We found significant differences in realism for the rattle and the dial lock; but not for the shaker (see Figure 15). Participants' ratings on the most realistic device clearly favored VoxelHap. Interestingly, for the Shaker the Vive controller came out as more enjoyable in the forced-choice question (see Figure 17). This is in line with participants' comments often stating that the VoxelHap proxy "feels too bulky for a salt shaker" (P5), "is difficult to hold because of its size" (P10) or "is heavier that I would expect it to be" (P2).

Participants were able to configure the resistance of safe Dial Lock's ($Mtorque = 1.0Nmm; SD = 0.4Nmm$). Since all selected values lie well within the possible Nmm range, we conclude that the RotationVoxel's technical capabilities to provide resistance is suitable for many rotational interactions, e.g., different knobs.

7.2.3 Experiment 1 & 2. Here, we descriptively compare both experiments, because a statistical comparison between them is not possible due to the iterative nature of our study design. First, when directly looking at participants' responses, it becomes evident that VoxelHap ratings were already quite high in experiment 1 (see Figure 16). Therefore, a distinct separation between ShapePlates vs. no ShapePlates was unlikely, especially given our sample size. Nevertheless, the questions regarding the most realistic and enjoyable device showed a clear increase in favor of ShapePlates, except for the Shaker. Here, adding ShapePlates led to a proxy that exceeded participants' expectations, both in terms of proxy size and weight. However, functionality, haptic feedback and shape still seemed to be convincing to receive relatively high realism scores.

ShapePlates are additional 3D-printed parts that can be attached to the Voxels. Yet it was unclear how this design choice might affect the perception of vibrotactile feedback. Therefore, we compared the configured strength (see Figure 18), synchronization and vibration scores between the two experiments (see Figure 16), but we observed no clear difference that would suggest a decrease in haptic resolution, potentially caused by damped vibrations.

7.2.4 Summary & Discussion. The results of experiment 2 confirm participants' positive responses towards VoxelHap proxies. Further, we found that the RotationVoxel's resistance feedback appeared to be in a reasonable range for hand interactions, and that attaching ShapePlates does not seem to affect vibrotactile resolution. Finally, we gathered insights into how the different levels of shape fidelity provided by VoxelHap affect users' perception. VoxelHap proxies with ShapePlates were selected as more realistic, even though the Shaker exceeded a reasonable size and weight. This may suggest that shape contributes more strongly to overall perception of realism than size [32] and weight. Nevertheless, it comes down to users' preferences and needs, which is in line with the core objective of VoxelHap, allowing users to approximate the (for them) crucial aspects, given the VR experience.

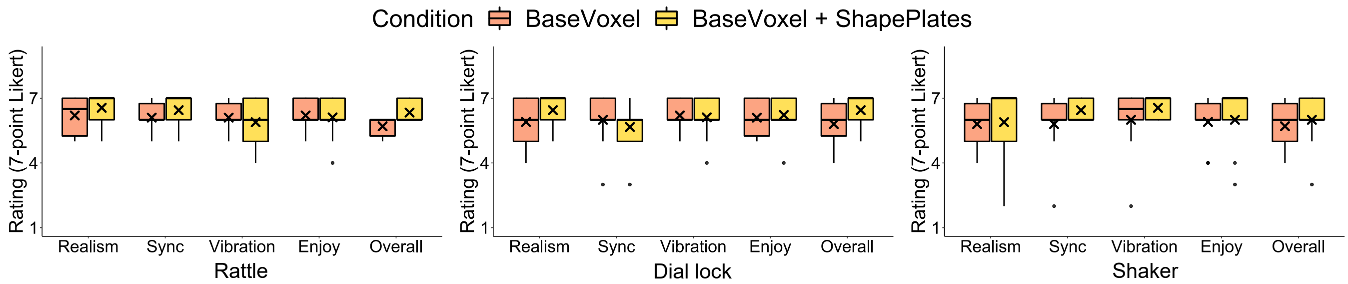

Figure 16: Experiment 1 and 2 descriptive comparison. VoxelHap proxies with vs. without ShapePlates.

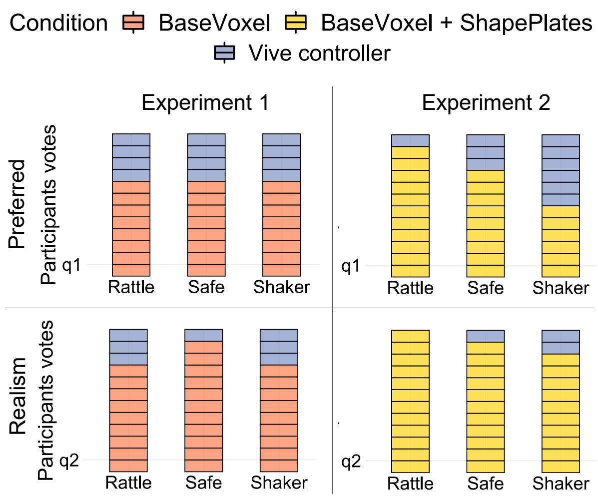

Figure 17: Participants' forced-choice votes regarding their preferred and most realistic device in experiment 1 and 2.

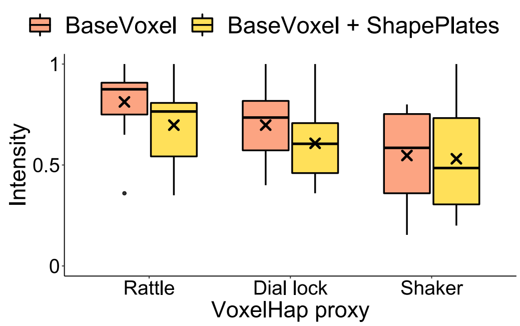

Figure 18: Participants' configured strength of vibrotactile feedback in experiment 1 and 2 using VoxelHap proxies.

8 DISCUSSION & LIMITATIONS

Finally, we discuss VoxelHap and its current limitations, and outline recommendations for future work.

8.1 VoxelHap Types of Haptic Feedback

As demonstrated above, VoxelHap proxies enable various types of haptic feedback such as simulating moving mass [10, 46, 56], stiffness [1, 4, 8], resistance [20] and size [7] variations or texture feedback [2, 12, 15]. In this work, we use application cases to evaluate the capabilities of VoxelHap, which together with our technical and user evaluation aligns with the proposed methods for toolkit evaluation according to Ledo et al. [34]. Yet, our user evaluation only included a subset of the demonstrated proxies. Thus, the created sensations need to be evaluated in future work.

To this end, we only built three types of Voxels, but VoxelHap is not limited to this. We also imagine integrated hinges [36], bendable [14, 23, 37], twistable [66] or stretchable [14, 17] parts to unlock more functionality and combinations of haptic feedback. Moreover, Voxels could also be equipped with wheels [55], allowing them to re-position themselves, acting as an encounter-type device [38]. There is also a possibility to include rich electrotactile feedback as suggest by Groeger et al. [22]. The authors embedded conductive pads and wires directly into the 3D-prints, providing tactile cues on various geometries. Since we use the same materials for fabrication, their technique could be adapted to further increase the haptic resolution of VoxelHap.

8.2 VoxelHap Beyond Virtual Reality

VoxelHap was designed for constructing proxies that can be used in VR, but any other XR technology could benefit from it. For example, functional physical visualizations are still used in many domains such as design or urban planning, because it is easier to understand spatial dimensions and facilitate discussions [26]. Users could collaboratively create functional drafts and the corresponding virtual model is generated automatically. Further, VoxelHap's rich input and output controls could be used for interactive music production or 3D animations by using a programming-by-demonstration paradigm [29]. We also envision VoxelHap proxies to be effective in remote collaboration, acting as a shared physical artifact. For example, the local novice could feel the manipulation of the proxy and could be guided through a task by a remote expert [18].

8.3 VoxelHap's Limitations

Given the technical capabilities of our Voxels, they do have a competitive size [14, 48, 60]. Nevertheless, they would benefit from an even more compact design. Both BaseVoxel and VibrationVoxel leave room for optimization; however, the RotationVoxel mainly determined final dimensions. This is a result of the chosen motor, because it needed to be large enough to produce human-noticeable resistance. We tested several different motor versions and opted for the smallest possible motor. This remains a problem of any active haptic device or component—they are often bulky, because of the forces they need to produce. Here, VR illusions could perhaps help to "suggest" rotational resistance [20].

Our construction pipeline only supports Voxels and passive ShapePlates. We decided against active ShapePlates, since it would have required us to equip ShapePlates with hardware, slowing down the fabrication and construction process. Depending on the use case, e.g., designers who re-use their ShapePlates in 3D-modeling applications [31, 35], some might benefit from active components. In addition, the construction process relies on the manually generated .hpdf which we aim to automate in the future. To this end, we did not evaluate user performance in constructing VoxelHap proxies but leave this for future work.

Each Voxel is a self-contained unit, and therefore comes with its own power source. This limits the scalability and usability to a certain extent. Voxels cannot share or distribute power and need to be recharged individually. One of our earlier prototypes used wireless charging, but this only works when the proxy is frequently placed and left on the table, for example encounter-type proxies. Another interesting approach would be to utilize users' manipulations of functional parts, i.e., rotation, to harvest energy [57].

9 CONCLUSION

In this paper we presented VoxelHap, a block-based construction toolkit to create functional proxies for VR. It consists of Voxels, blocks with special functionalities, and Plates, to increase the haptic resolution. It is the first VR toolkit that incorporates VR illusion techniques into the design. We presented a range of fully functional proxies, for various use cases and applications, illustrating the potential of the toolkit. Our technical evaluation ought to help assess its capabilities and limitations. In two experiments with 24 participants, we evaluate a subset of the constructed proxies and study how they compare to a traditional VR controller. First, we investigated VoxelHap's combined haptic feedback, and second, the trade-offs of VoxelHap's ShapePlates. Our findings show that VoxelHap's proxies outperform traditional controllers and were generally favored by participants. Finally, by open-sourcing VoxelHap's hardware, we hope to ease access, save resources and encourage the community to contribute to haptic interfaces—even beyond VR.

ACKNOWLEDGMENTS

This research was funded by the German Federal Ministry of Education and Research (BMBF) under grant number 01IS17043 (project VRGiO). We would also like to thank Hanna Kriegler, Jan Dickmann, Juliana Keller and Shahrukh Khan for their early contributions to this work.

REFERENCES

[1] Adilzhan Adilkhanov, Amir Yelenov, Ramakanth Singal Reddy, Alexander Terekhov, and Zhanat Kappassov. 2020. VibeRo: Vibrotactile Stiffness Perception Interface for Virtual Reality. *IEEE Robotics and Automation Letters* 5, 2 (2020), 2785–2792. https://doi.org/10.1109/LRA.2020.2972793

[2] Bruno Araujo, Ricardo Jota, Varun Perumal, Jia Xian Yao, Karan Singh, and Daniel Wigdor. 2016. Snake Charmer: Physically Enabling Virtual Objects. In *Proceedings of the TEI '16: Tenth International Conference on Tangible, Embedded, and Embodied Interaction* (Eindhoven, Netherlands) *(TEI '16)*. Association for Computing Machinery, New York, NY, USA, 218–226. https://doi.org/10.1145/2839462.2839484

[3] Jatin Arora, Aryan Saini, Nirmita Mehra, Varnit Jain, Shwetank Shrey, and Aman Parnami. 2019. VirtualBricks: Exploring a Scalable, Modular Toolkit for Enabling Physical Manipulation in VR. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300286

[4] Yuki Ban, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. 2014. Controlling Perceived Stiffness of Pinched Objects using Visual Feedback of Hand Deformation. In *2014 IEEE Haptics Symposium (HAPTICS)*. 557–562. https://doi.org/10.1109/HAPTICS.2014.6775516

[5] Yuki Ban, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. 2014. Displaying Shapes with Various Types of Surfaces Using Visuo-Haptic Interaction. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology* (Edinburgh, Scotland) *(VRST '14)*. Association for Computing Machinery, New York, NY, USA, 191–196. https://doi.org/10.1145/2671015.2671028

[6] Christopher C Berger, Mar Gonzalez-Franco, Eyal Ofek, and Ken Hinckley. 2018. The Uncanny Valley of Haptics. *Science Robotics* 3, 17 (2018), eaar7010.

[7] Joanna Bergström, Aske Mottelson, and Jarrod Knibbe. 2019. Resized Grasping in VR: Estimating Thresholds for Object Discrimination. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1175–1183. https://doi.org/10.1145/3332165.3347939

[8] Elodie Bouzbib, Claudio Pacchierotti, and Anatole Lécuyer. 2023. When Tangibles Become Deformable: Studying Pseudo-Stiffness Perceptual Thresholds in a VR Grasping Task. *IEEE Transactions on Visualization and Computer Graphics* (2023), 1–10. https://doi.org/10.1109/TVCG.2023.3247083

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

[10] Inrak Choi, Heather Culbertson, Mark R. Miller, Alex Olwal, and Sean Follmer. 2017. Grabity: A Wearable Haptic Interface for Simulating Weight and Grasping in Virtual Reality. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 119–130. https://doi.org/10.1145/3126594.3126599

[11] Donald Degraen, Bruno Fruchard, Frederik Smolders, Emmanouil Potetsianakis, Seref Güngör, Antonio Krüger, and Jürgen Steimle. 2021. Weirding Haptics: In-Situ Prototyping of Vibrotactile Feedback in Virtual Reality through Vocalization. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 936–953. https://doi.org/10.1145/3472749.3474797

[12] Donald Degraen, André Zenner, and Antonio Krüger. 2019. Enhancing Texture Perception in Virtual Reality Using 3D-Printed Hair Structures. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland UK) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1—-12. https://doi.org/10.1145/3290605.3300479

[13] Tim Düwel, Martin Feick, and Antonio Krüger. 2021. Considering Interaction Types for Geometric Primitive Matching. In *Everyday Proxy Objects for Virtual Reality Workshop of the 2021 CHI Conference on Human Factors in Computing Systems*. http://epo4vr.dfki.de/

[14] Martin Feick, Scott Bateman, Anthony Tang, André Miede, and Nicolai Marquardt. 2020. Tangi: Tangible Proxies for Embodied Object Exploration and Manipulation in Virtual Reality. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 195–206. https://doi.org/10.1109/ISMAR50242.2020.00042

[15] Martin Feick, Donald Degraen, Fabian Hupperich, and Antonio Krüger. 2023. MetaReality: Enhancing Tactile Experiences using Actuated 3D-printed Metamaterials in Virtual Reality. *Frontiers in Virtual Reality* 4 (2023). https://doi.org/10.3389/frvir.2023.1172381

[16] Martin Feick, Niko Kleer, Anthony Tang, and Antonio Krüger. 2020. The Virtual Reality Questionnaire Toolkit. In *Adjunct Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology, AP UIST 2020* (Virtual Event, USA) *(UIST '20 Adjunct)*. ACM, 68–69. https://dl.acm.org/doi/10.1145/3379350.3416188

[17] Martin Feick, Niko Kleer, André Zenner, Anthony Tang, and Antonio Krüger. 2021. Visuo-Haptic Illusions for Linear Translation and Stretching Using Physical Proxies in Virtual Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 220, 13 pages. https://doi.org/10.1145/3411764.3445456

[18] Martin Feick, Terrance Mok, Anthony Tang, Lora Oehlberg, and Ehud Sharlin. 2018. Perspective on and Re-Orientation of Physical Proxies in Object-Focused Remote Collaboration. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173855

[19] Martin Feick, Kora Persephone Regitz, Anthony Tang, and Antonio Krüger. 2022. Designing Visuo-Haptic Illusions with Proxies in Virtual Reality: Exploration of Grasp, Movement Trajectory and Object Mass. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 635, 15 pages. https://doi.org/10.1145/3491102.3517671

[20] Martin Feick, André Zenner, Oscar Javier Ariza Nunez, Anthony Tang, Biyikli Cihan, and Antonio Krüger. 2023. Turn-It-Up: Rendering Resistance for Knobs in Virtual Reality through Undetectable Pseudo-Haptics. In *Proceedings of the 36nd Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3586183.3606722

[21] James J. Gibson. 1933. Adaption, After-effect and Contrast in the Perception of Curved Lines. *Journal of Experimental Psychology* 16, 1 (1933), 1–31. https://doi.org/10.1037/h0074626

[22] Daniel Groeger, Martin Feick, Anusha Withana, and Jürgen Steimle. 2019. Tactlets: Adding Tactile Feedback to 3D Objects Using Custom Printed Controls. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 923–936. https://doi.org/10.1145/3332165.3347937

[23] Seongkook Heo, Jaeyeon Lee, and Daniel Wigdor. 2019. PseudoBend: Producing Haptic Illusions of Stretching, Bending, and Twisting Using Grain Vibrations. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 803–813. https://doi.org/10.1145/3332165.3347941

[24] Anuruddha Hettiarachchi and Daniel Wigdor. 2016. 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* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1957–1967. https://doi.org/10.1145/2858036.2858134

[25] Ken Hinckley, Randy Pausch, John C. Goble, and Neal F. Kassell. 1994. Passive Real-World Interface Props for Neurosurgical Visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, Massachusetts, USA) *(CHI '94)*. Association for Computing Machinery, New York, NY, USA, 452–458. https://doi.org/10.1145/191666.191821

[26] Carmen Hull and Wesley Willett. 2017. Building with Data: Architectural Models as Inspiration for Data Physicalization. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1217–1264. https://doi.org/10.1145/3025453.3025850

[27] Brent Edward Insko. 2001. *Passive Haptics Significantly Enhances Virtual Environments*. Ph. D. Dissertation. The University of North Carolina at Chapel Hill, Chapel Hill. http://www.cs.unc.edu/techreports/01-017.pdf

[28] Seunggon Jeon, Seungwon Paik, Ungyeon Yang, Patrick C Shih, and Kyungsik Han. 2021. The More, the Better? Improving VR Firefighting Training System with Realistic Firefighter Tools as Controllers. *Sensors (Basel, Switzerland)* 21, 21 (2021), 7193.

[29] Sergi Jordà, Günter Geiger, Marcos Alonso, and Martin Kaltenbrunner. 2007. The ReacTable: Exploring the Synergy between Live Music Performance and Tabletop Tangible Interfaces. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction* (Baton Rouge, Louisiana) *(TEI '07)*. Association for Computing Machinery, New York, NY, USA, 139–146. https://doi.org/10.1145/1226969.1226998

[30] Johan Kildal. 2012. Kooboh: Variable Tangible Properties in a Handheld Haptic-Illusion Box. In *Proceedings of the 2012 International Conference on Haptics: Perception, Devices, Mobility, and Communication - Volume Part II* (Tampere, Finland) *(EuroHaptics'12)*. Springer-Verlag, Berlin, Heidelberg, 191–194. https://doi.org/10.1007/978-3-642-31404-9_33

[31] Yoshifumi Kitamura, Yuichi Itoh, and Fumio Kishino. 2001. Real-Time 3D Interaction with ActiveCube. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems* (Seattle, Washington) *(CHI EA '01)*. Association for Computing Machinery, New York, NY, USA, 355–356. https://doi.org/10.1145/634067.634277

[32] Eun Kwon, Gerard J. Kim, and Sangyoon Lee. 2009. Effects of sizes and shapes of props in tangible augmented reality. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*. 201–202. https://doi.org/10.1109/ISMAR.2009.5336463

[33] Anatole Lécuyer. 2009. Simulating haptic feedback using vision: A survey of research and applications of pseudo-haptic feedback. *Presence: Teleoperators and Virtual Environments* 18, 1 (2009), 39–53.

[34] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. 2018. Evaluation Strategies for HCI Toolkit Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–17. https://doi.org/10.1145/3173574.3173610

[35] Danny Leen, Raf Ramakers, and Kris Luyten. 2017. StrutModeling: A Low-Fidelity Construction Kit to Iteratively Model, Test, and Adapt 3D Objects. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) *(UIST '17)*. Association for Computing Machinery, New York, NY, USA, 471–479. https://doi.org/10.1145/3126594.3126643

[36] Nianlong Li, Han-Jong Kim, LuYao Shen, Feng Tian, Teng Han, Xing-Dong Yang, and Tek-Jin Nam. 2020. HapLinkage: Prototyping Haptic Proxies for Virtual Hand Tools Using Linkage Mechanism. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 1261–1274. https://doi.org/10.1145/3379337.3415812

[37] John C. McClelland, Robert J. Teather, and Audrey Girouard. 2017. Haptobend: Shape-Changing Passive Haptic Feedback in Virtual Reality. In *Proceedings of the 5th Symposium on Spatial User Interaction* (Brighton, United Kingdom) *(SUI '17)*. Association for Computing Machinery, New York, NY, USA, 82–90. https://doi.org/10.1145/3131277.3132179

[38] William A. McNeely. 1993. Robotic Graphics: A New Approach to Force Feedback for Virtual Reality. In *Proceedings of IEEE Virtual Reality Annual International Symposium (VRAIS '93)*. 336–341. https://doi.org/10.1109/VRAIS.1993.380761

[39] Stefanie Mueller, Tobias Mohr, Kerstin Guenther, Johannes Frohnhofen, and Patrick Baudisch. 2014. FaBrickation: Fast 3D Printing of Functional Objects by Integrating Construction Kit Building Blocks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 3827–3834. https://doi.org/10.1145/2556288.2557005

[40] Thomas Muender, Anke V. Reinschluessel, Sean Drewes, Dirk Wenig, Tanja Döring, and Rainer Malaka. 2019. Does It Feel Real? Using Tangibles with Different Fidelities to Build and Explore Scenes in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland, UK) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300903

[41] Niels Christian Nilsson, Andre Zenner, and Adalberto L. Simeone. 2021. Propping Up Virtual Reality With Haptic Proxies. *IEEE Computer Graphics and Applications* 41, 5 (Sept. 2021), 104–112. https://doi.org/10.1109/MCG.2021.3097671

[42] Tommy Nilsson, Flavie Rometsch, Leonie Becker, Florian Dufresne, Paul Demedeiros, Enrico Guerra, Andrea Emanuele Maria Casini, Anna Vock, Florian Gaeremynck, and Aidan Cowley. 2023. Using Virtual Reality to Shape Humanity's Return to the Moon: Key Takeaways from a Design Study. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 305, 16 pages. https://doi.org/10.1145/3544548.3580718

[43] Martin Nisser, Leon Cheng, Yashaswini Makaram, Ryo Suzuki, and Stefanie Mueller. 2022. ElectroVoxel: Electromagnetically Actuated Pivoting for Scalable Modular Self-Reconfigurable Robots. In *2022 International Conference on Robotics and Automation (ICRA)*. 4254–4260. https://doi.org/10.1109/ICRA46639.2022.9811746

[44] Chaeyong Park, Jinhyuk Yoon, Seungjae Oh, and Seungmoon Choi. 2020. Augmenting Physical Buttons with Vibrotactile Feedback for Programmable Feels. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 924–937. https://doi.org/10.1145/3379337.3415837

[45] A. Roudaut, D. Krusteva, M. McCoy, A. Karnik, K Ramani, and S. Subramanian. 2016. Cubimorph: Designing modular interactive devices. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*. 3339–3345. https://doi.org/10.1109/ICRA.2016.7487508

[46] Shahabedin Sagheb, Frank Wencheng Liu, Alireza Bahremand, Assegid Kidane, and Robert LiKamWa. 2019. SWISH: A Shifting-Weight Interface of Simulated Hydrodynamics for Haptic Perception of Virtual Fluid Vessels. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 751–761. https://doi.org/10.1145/3332165.3347870

[47] Majed Samad, Elia Gatti, Anne Hermes, Hrvoje Benko, and Cesare Parise. 2019. Pseudo-Haptic Weight: Changing the Perceived Weight of Virtual Objects by Manipulating Control-Display Ratio. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300550

[48] Philipp Schoessler, Daniel Windham, Daniel Leithinger, Sean Follmer, and Hiroshi Ishii. 2015. Kinetic Blocks: Actuated Constructive Assembly for Interaction and Display. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 341–349. https://doi.org/10.1145/2807442.2807453

[49] Eric Schweikardt and Mark D Gross. 2008. The Robot is the Program: Interacting with RoBlocks. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction* (Bonn, Germany) *(TEI '08)*. Association for Computing Machinery, New York, NY, USA, 167–168. https://doi.org/10.1145/1347390.1347427

[50] Adalberto L. Simeone, Eduardo Velloso, and Hans Gellersen. 2015. 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* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 3307–3316. https://doi.org/10.1145/2702123.2702389

[51] Alexa F. Siu, Eric J. Gonzalez, Shenli Yuan, Jason B. Ginsberg, and Sean Follmer. 2018. 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* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3173574.3173865

[52] Antun Skuric, Hasan Sinan Bank, Richard Unger, Owen Williams, and David González-Reyes. 2022. SimpleFOC: A Field Oriented Control (FOC) Library for Controlling Brushless Direct Current (BLDC) and Stepper Motors. *Journal of Open Source Software* 7, 74 (2022), 4232. https://doi.org/10.21105/joss.04232

[53] Patrick L. Strandholt, Oana A. Dogaru, Niels C. Nilsson, Rolf Nordahl, and Stefania Serafin. 2020. 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* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376303

[54] Paul Strohmeier and Kasper Hornbæk. 2017. Generating Haptic Textures with a Vibrotactile Actuator. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4994–5005. https://doi.org/10.1145/3025453.3025812

[55] Ryo Suzuki, Eyal Ofek, Mike Sinclair, Daniel Leithinger, and Mar Gonzalez-Franco. 2021. 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* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 1269–1281. https://doi.org/10.1145/3472749.3474821

[56] Yudai Tanaka, Arata Horie, and Xiang 'Anthony' Chen. 2020. DualVib: Simulating Haptic Sensation of Dynamic Mass by Combining Pseudo-Force and Texture Feedback. In *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology* (Virtual Event, Canada) *(VRST '20)*. Association for Computing Machinery, New York, NY, USA, Article 1, 10 pages. https://doi.org/10.1145/3385956.3418964

[57] Shan-Yuan Teng, K. D. Wu, Jacqueline Chen, and Pedro Lopes. 2022. Prolonging VR Haptic Experiences by Harvesting Kinetic Energy from the User. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 39, 18 pages. https://doi.org/10.1145/3526113.3545635

[58] Xavier de Tinguy, Claudio Pacchierotti, Mathieu Emily, Mathilde Chevalier, Aurélie Guignardat, Morgan Guillaudeux, Chloé Six, Anatole Lécuyer, and Maud Marchal. 2019. How Different Tangible and Virtual Objects Can Be While Still Feeling the Same?. In *2019 IEEE World Haptics Conference (WHC)*. 580–585. https://doi.org/10.1109/WHC.2019.8816164

[59] Hsin-Ruey Tsai, Ching-Wen Hung, Tzu-Chun Wu, and Bing-Yu Chen. 2020. ElastOscillation: 3D Multilevel Force Feedback for Damped Oscillation on VR Controllers. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376408

[60] Ryoichi Watanabe, Yuichi Itoh, Masatsugu Asai, Yoshifumi Kitamura, Fumio Kishino, and Hideo Kikuchi. 2004. The Soul of ActiveCube: Implementing a Flexible, Multimodal, Three-Dimensional Spatial Tangible Interface. *Comput. Entertain.* 2, 4 (oct 2004), 15. https://doi.org/10.1145/1037851.1037874

[61] Jackie (Junrui) Yang, Hiroshi Horii, Alexander Thayer, and Rafael Ballagas. 2018. VR Grabbers: Ungrounded Haptic Retargeting for Precision Grabbing Tools. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 889–899. https://doi.org/10.1145/3242587.3242643

[62] Shigeo Yoshida, Yuqian Sun, and Hideaki Kuzuoka. 2020. PoCoPo: Handheld Pin-Based Shape Display for Haptic Rendering in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376358

[63] André Zenner and Antonio Krüger. 2017. Shifty: A Weight-Shifting Dynamic Passive Haptic Proxy to Enhance Object Perception in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics* 23, 4 (2017), 1285–1294. https://doi.org/10.1109/TVCG.2017.2656978

[64] André Zenner, Kristin Ullmann, and Antonio Krüger. 2021. Combining Dynamic Passive Haptics and Haptic Retargeting for Enhanced Haptic Feedback in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2627–2637. https://doi.org/10.1109/TVCG.2021.3067777

[65] Yiwei Zhao, Lawrence H. Kim, Ye Wang, Mathieu Le Goc, and Sean Follmer. 2017. 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* (Brighton, United Kingdom) *(ISS '17)*. Association for Computing Machinery, New York, NY, USA, 82–91. https://doi.org/10.1145/3132272.3134143

[66] Kening Zhu, Taizhou Chen, Feng Han, and Yi-Shiun Wu. 2019. HapTwist: Creating Interactive Haptic Proxies in Virtual Reality Using Low-Cost Twistable Artefacts. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300923