



Haplets: Finger-Worn Wireless and Low-Encumbrance Vibrotactile Haptic Feedback for Virtual and Augmented Reality

Pornthep Preechayasomboon^{1*} and Eric Rombokas^{1,2}

¹Rombolabs, Mechanical Engineering, University of Washington, Seattle, WA, United States, ²Electrical Engineering, University of Washington, Seattle, WA, United States

We introduce Haplets, a wearable, low-encumbrance, finger-worn, wireless haptic device that provides vibrotactile feedback for hand tracking applications in virtual and augmented reality. Haplets are small enough to fit on the back of the fingers and fingernails while leaving the fingertips free for interacting with real-world objects. Through robust physically-simulated hands and low-latency wireless communication, Haplets can render haptic feedback in the form of impacts and textures, and supplements the experience with pseudo-haptic illusions. When used in conjunction with handheld tools, such as a pen, Haplets provide haptic feedback for otherwise passive tools in virtual reality, such as for emulating friction and pressure-sensitivity. We present the design and engineering for the hardware for Haplets, as well as the software framework for haptic rendering. As an example use case, we present a user study in which Haplets are used to improve the line width accuracy of a pressure-sensitive pen in a virtual reality drawing task. We also demonstrate Haplets used during manipulation of objects and during a painting and sculpting scenario in virtual reality. Haplets, at the very least, can be used as a prototyping platform for haptic feedback in virtual reality.

OPEN ACCESS

Edited by:

Daniel Leithinger,
University of Colorado Boulder,
United States

Reviewed by:

Pedro Lopes,
University of Chicago, United States
Ken Nakagaki,
Massachusetts Institute of
Technology, United States

*Correspondence:

Pornthep Preechayasomboon
pnrthp@uw.edu

Specialty section:

This article was submitted to
Haptics,
a section of the journal
Frontiers in Virtual Reality

Received: 09 July 2021

Accepted: 06 September 2021

Published: 20 September 2021

Citation:

Preechayasomboon P and
Rombokas E (2021) Haplets: Finger-
Worn Wireless and Low-Encumbrance
Vibrotactile Haptic Feedback for Virtual
and Augmented Reality.
Front. Virtual Real. 2:738613.
doi: 10.3389/frvir.2021.738613

Keywords: haptics, virtual reality, augmented reality, spatial computing, sensory feedback, human computer interface

1 INTRODUCTION

Hands can be considered the most dexterous tool that a human naturally possesses, making them the most obvious input modality for virtual reality (VR) and augmented reality (AR). In productivity tasks in AR and VR, natural hand tracking enables seamless context switching between the virtual and physical world i.e., not having to put down a controller first to interact with a physical keyboard. However, a major limitation is the lack of haptic feedback that leads to a poor experience in scenarios that require manual dexterity such as object manipulation, drawing, writing, and typing on a virtual keyboard (Gupta et al., 2020). Using natural hand input invokes the visuo-haptic neural representations of held objects when they are seen and felt in AR and VR (Lengyel et al., 2019). Although haptic gloves seem promising for rendering a realistic sense of touch and textures, or provide kinesthetic impedance in the virtual or augmented space, wearing a glove that covers the fingers greatly reduces the tactile information from physical objects outside the augmented space. Thus having a solution that provides believable haptic feedback with the lowest encumbrance is desirable.

Numerous research devices have shown that there is value in providing rich haptic feedback to the fingertips during manipulation (Johansson and Flanagan, 2009; Schorr and Okamura, 2017; Hinchet et al., 2018; Lee et al., 2019), texture perception (Chan et al., 2021), stiffness perception (Salazar et al., 2020), and normal and shear force perception (Kim et al., 2018; Preechayasomboon et al., 2020). Although these devices may render high fidelity haptic feedback, they often come at the cost of being tethered to another device or have bulky electronics that impede the wearability of the device and ultimately hinder immersion of the VR experience. Additionally, once devices are placed on the fingertips, any interaction with objects outside the virtual space is rendered impossible unless the device is removed or put down. Teng et al. (2021) has shown that wearable, wireless, low encumbrance haptic feedback on the fingertips is useful for AR scenarios with a prototype that leaves the fingertips free when haptic feedback is not required. Akin to the growing adoption of virtual reality, the device must be as frictionless to the user as possible—wearable haptic devices are no exception.

It has been shown that rendering haptic feedback away from the intended site does provide meaningful sensations that can be interpreted as proxies for the interactions at the hand (Pezent et al., 2019), or for mid-air text entry (Gupta et al., 2020). Ando et al. (2007) has shown that rendering vibrations on the fingernail can be used to augment passive touch-sensitive displays for creating convincing perception of edges and textures, others have extended this technique to include projector-based augmented reality (Rekimoto, 2009), and even used the fingernail as a haptic display itself (Hsieh et al., 2016). We have shown that there is a perceptual tolerance for conflicting locations of visual and tactile touch, in which the two sensory modalities are fused into a single percept despite arising from different locations (Caballero and Rombokas, 2019). Furthermore, combining multiple modalities either in the form of augmenting otherwise passive haptic sensations (Choi et al., 2020), using pseudo-haptic illusions (Achibet et al., 2017; Samad et al., 2019), or a believable simulation (Kuchenbecker et al., 2006; Chan et al., 2021), can possibly mitigate the lack of congruence between the visual and tactile sensation. We therefore extend what Ando et al. (2007) has proposed to immersive virtual reality by placing the haptic device on the fingernail and finger dorsum and compensating for the distant stimulation with believable visual and haptic rendering, which leaves the fingerpads still free to interact with real-world objects.

With the hands now free to hold and interact with physical objects, any passive object can become a tangible prop or tool. These held tools can provide passive haptic feedback while presenting familiar grounding and pose for the fingers. Gripmarks (Zhou et al., 2020) has shown that everyday objects can be used as mixed reality input by using the hand's pose as an estimate to derive the object being held. In this paper, we further this concept by introducing Haplets: small, wireless and wearable haptic actuators. Each Haplet is a self-contained unit that consists of the bare minimum required to render vibrotactile stimulus wirelessly: a linear resonant actuator (LRA), a motor driver, a wireless system-on-a-chip (SoC), and a battery. Haplets are worn

on the dorsal side of the finger and fingernail, and have a footprint small enough that the hands can still be tracked using computer vision methods. Combined with a believable simulation for rendering vibrotactile feedback in VR, Haplets can be used to augment the sensation of manipulation, textures and stiffness for bare hands while still maintaining the ability to pick up and handle everyday objects outside the virtual space. With a tool held in the hand, Haplets can render haptic effects to emulate the sensations when the tool interacts with the virtual environment. We use Haplets as an exploration platform towards building low-encumbrance, wearable haptic feedback devices for virtual and augmented reality.

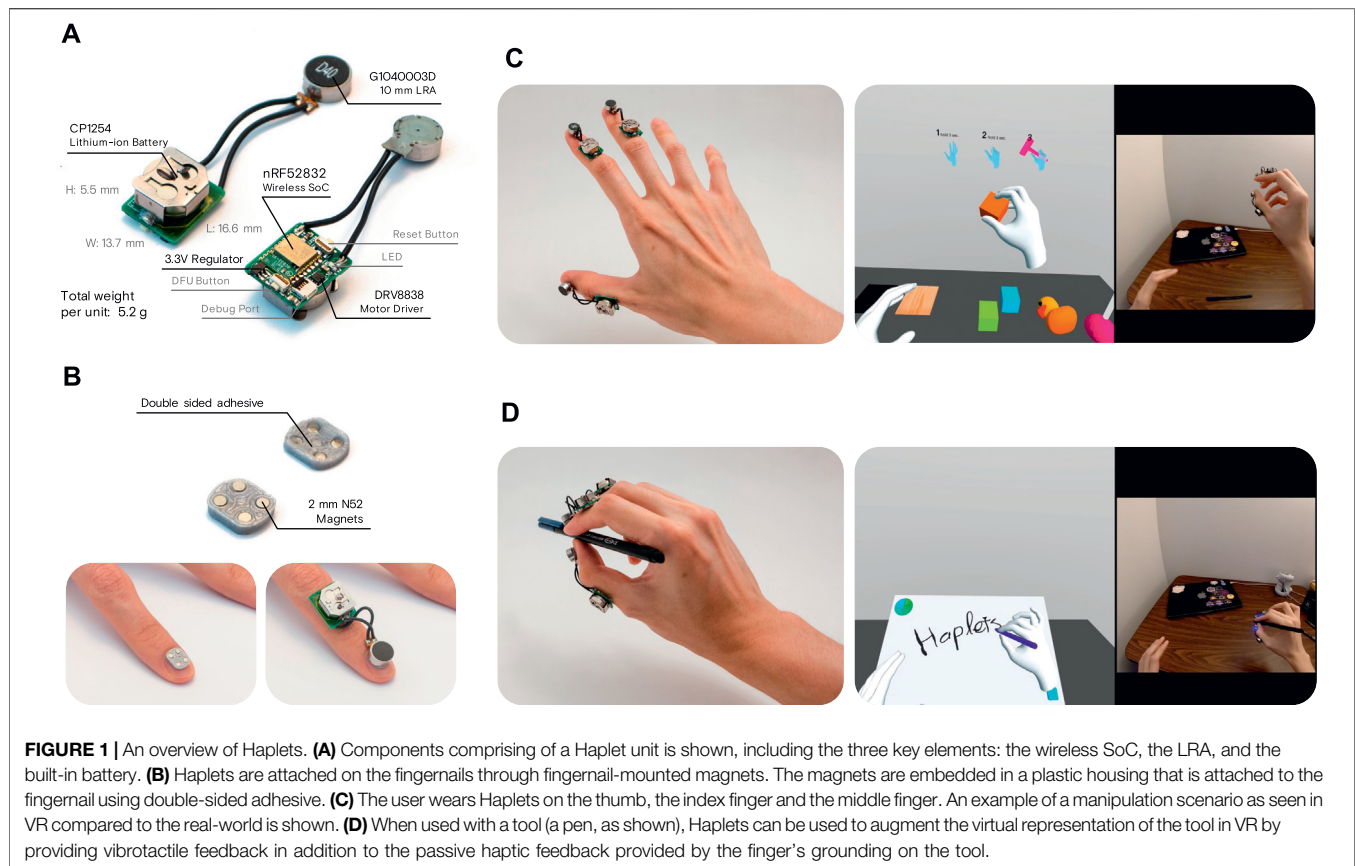
The rest of this paper is organized as follows: first, in **Section 2**, we describe the hardware for each Haplet and engineering choices made for each component, including our low-latency wireless communication scheme. We then briefly cover the characterization efforts for the haptic actuator (the LRA). Then, we cover our software efforts in creating a physically-believable virtual environment that drives our haptic experiences, including physics-driven virtual hands and augmented physical tools. In **Section 3**, we cover a small user study to highlight one use case of Haplets and explore the practicality of Haplets in a virtual reality scenario. In **Section 4**, we demonstrate other use cases for Haplets in virtual or augmented reality environment such as manipulation, texture discrimination, and painting with tools. Finally, in **Section 5**, we discuss our engineering efforts and the results of our user study, and provide insight for shortcomings and potential future work.

2 MATERIALS AND METHODS

Haplets can be thought of as distributed wireless wearable haptic actuators. As mentioned previously, each Haplet consists of the bare minimum required to render haptic effects: an LRA, a motor driver, a wireless SoC, and a battery (**Figure 1A**). We minimized the footprint so that Haplets, aside from our target area of the finger, can be worn on other parts of the body such as the wrist, arms, or face, or integrated into other, larger systems. Haplets is designed to be able to drive other voice-coil based actuators such as voice coil motors (VCMs), eccentric rotating mass (ERM) actuators, and small brushed DC motors, as well.

2.1 Finger-Mounted Hardware

The core electronic components of each Haplet, as shown in **Figure 1A**, are contained within one side of 13.7 mm by 16.6 mm PCB, while the other side of the PCB is a coin cell socket. We use a BC832 wireless module (Fanstel) that consists of a nRF52832 SoC (Nordic Semiconductor) and an integrated radio antenna. The motor driver for the LRA is a DRV8838 (Texas Instruments), which is chosen for its high frequency, non-audible, pulse width modulation limit (at 250 kHz) and versatile voltage input range (from 0 to 11 V). The PCB also consists of a J-Link programming and debug port, light emitting diode (LED), and two tactile buttons for resetting the device and entering device firmware upgrade (DFU) mode. The DFU mode is used for programming Haplets over a Bluetooth connection. The coin cell we use is a



Lithium-ion CP1254 (Varta), measuring 12 mm in diameter by 5.4 mm in height, which is chosen for its high current output (120 mA) and high power density. In our tests, Haplets can be used for up to 3 h of typical usage and the batteries can be quickly replaced. The total weight of one Haplet unit, including the LRA, is 5.2 g.

We imagine Haplets as a wearable device, therefore Haplets must be able to be donned and doffed with minimal effort. To achieve this, we use 3D printed nail covers with embedded magnets, as shown in **Figure 1B**, to attach the LRA to the fingernail. The nail covers are small, lightweight, and can be attached to the fingernail using double-sided adhesive. Each cover has a concave curvature that corresponds to each fingernail. The Haplets' PCB is attached to the dorsal side of the middle phalanx using silicone-based, repositionable double-sided adhesive tape. In our user studies and demonstrations, we place the Haplets on the thumb, index finger and middle finger of the right hand, as shown in **Figure 1C**.

2.2 Low-Latency Wireless Communication

Since we target the fingers, we desire to reduce the latency from visual perception to tactile perception as much as possible, especially when considering the mechanical time constant of the LRA¹. Although Bluetooth Low Energy (BLE) is

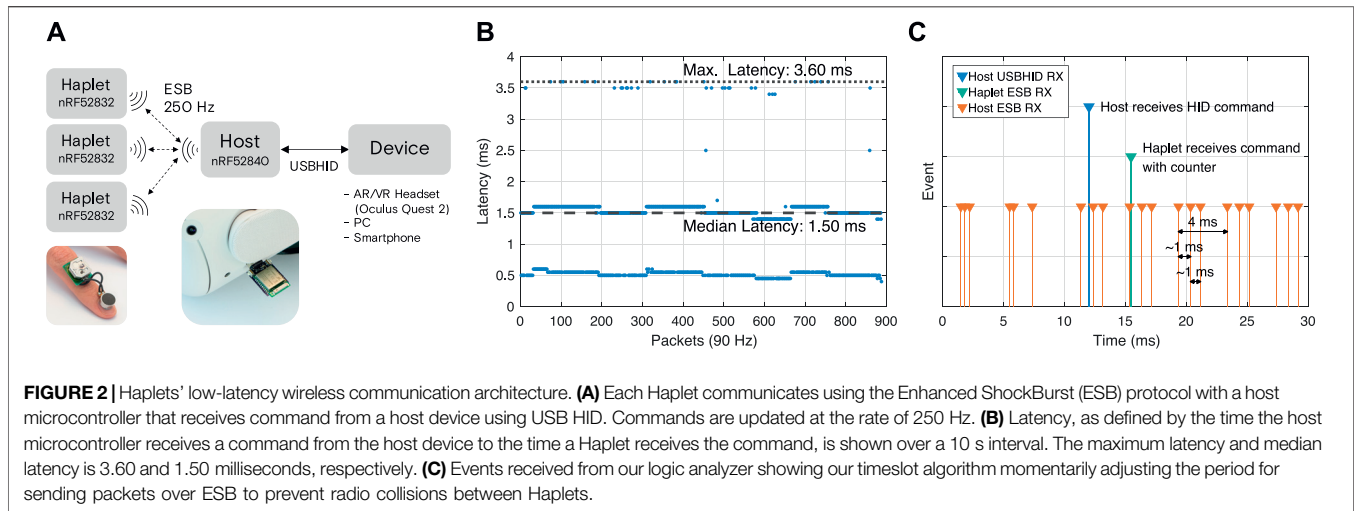
commonplace and readily available in most systems with a wireless interface, the overall latency can vary from device to device. We therefore opted to use Enhanced ShockBurst (ESB)², a proprietary radio protocol developed by Nordic Semiconductor, for our devices instead. ESB enables up to eight primary transmitters³ to communicate with a primary receiver. In our implementation, each Haplet is a primary transmitter that sends a small packet at a fixed interval to a host microcontroller, a primary receiver, which is another SoC that is connected to a VR headset, smartphone or PC (**Figure 2A**). Commands for each Haplet are sent in return along with the acknowledge (ACK) packet from the host device to the Haplets. If each Haplet transmits at an interval of 4 milliseconds, then ideally the maximum latency will be slightly over 4 milliseconds when accounting for radio transmission times for the ACK packet.

Since Haplets transmit at a high frequency (every 4 ms or 250 Hz), there is a high chance of collisions between multiple units. We mitigate this by employing a simple time-slot synchronization scheme between the Haplets and the host

²https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/ug_esb.html

³It is worth noting that although this suggests that a maximum of eight Haplets can be communicating with one host microcontroller at once, there exists techniques such as radio time-slot synchronization similar to those used in Bluetooth that can increase the number of concurrent transmitters to 20.

¹<https://www.vibration-motor.com/wp-content/uploads/2019/05/G1040003D.pdf>



microcontroller where each Haplet must transmit in its own predefined 500 microsecond timeslot. The host microcontroller keeps a 250 Hz clock and a microsecond counter that resets every tick of the 250 Hz clock. The counter value from the host is transmitted along with the command packets and each Haplet then uses the counter value to adjust its next transmitting interval to correct itself. For instance, if a Haplet receives a counter value of 750 microseconds and its timeslot is at 1,000 microseconds, it will delay its next round of transmission by 250 microseconds or 4,250 microseconds in total, after the correction, it will transmit at the usual 4,000 microsecond interval until another correction is needed.

One drawback of our implementation, as we use a proprietary radio protocol, is we cannot use the built-in Bluetooth capabilities of host devices (i.e., VR headsets or PC) to communicate with Haplets. Therefore, we use a nRF52840 SoC (Nordic Semiconductor) as a host microcontroller that communicates with the host device through a wired USB connection. In order to minimize the end-to-end latency, we use the Human Interface Device (HID) class for our USB connection. The benefits are twofold: 1. HID has a typical latency of 1 millisecond and 2. HID is compatible out-of-the-box with most modern hardware including both Windows and Unix-based PCs, standalone VR headsets such as the Oculus Quest, and most Android-based devices (Preechayasomboon et al., 2020).

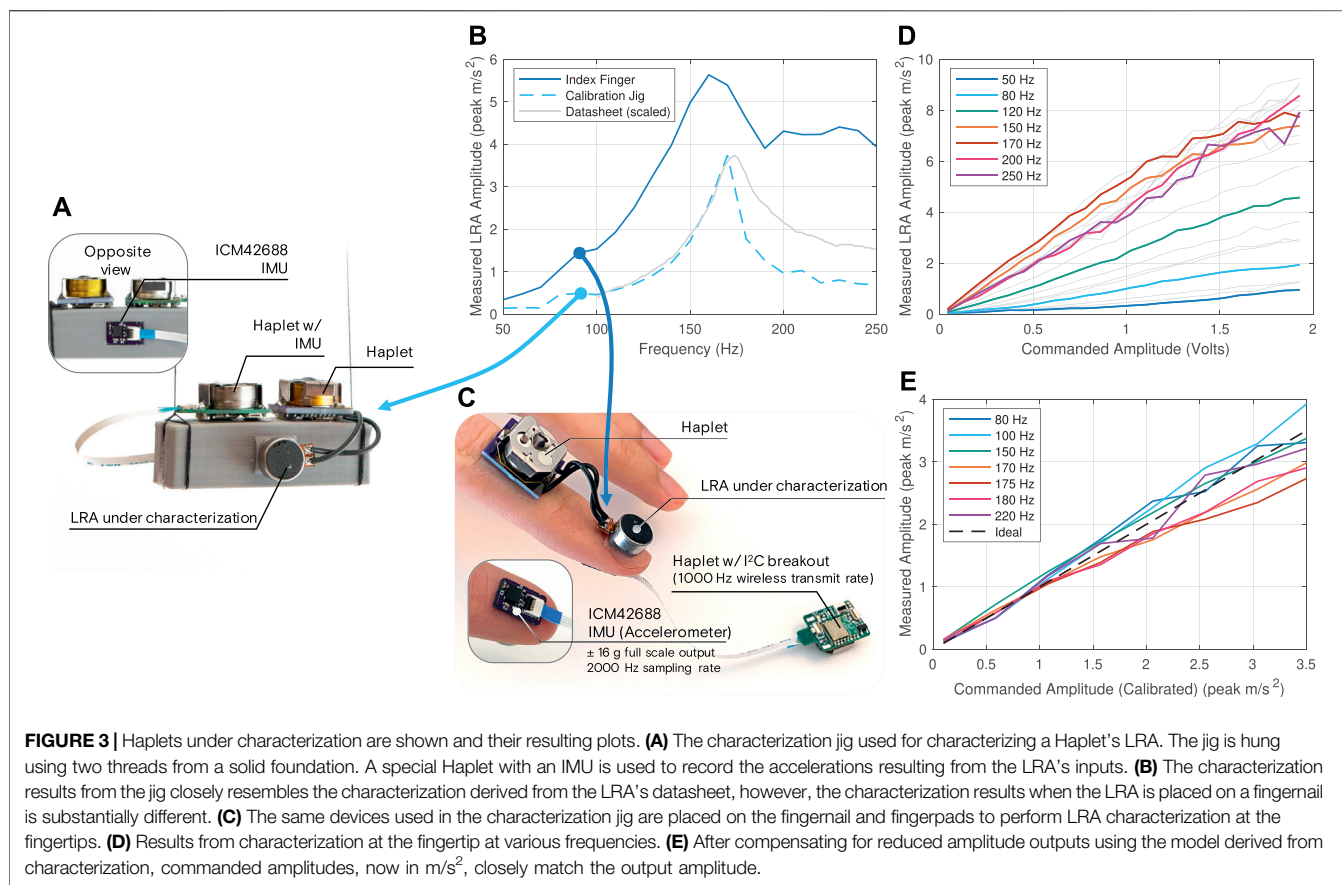
We briefly tested the communication latency of our system by running a test program that sends command packets over HID to our host microcontroller to three Haplets at 90 Hz—this frequency is chosen to simulate the typical framerate for VR applications. Two digital output pins, one from a Haplet and one from the host microcontroller, were connected to a logic analyzer. The Haplet's output pin toggles when a packet is received and the host microcontroller's output pin toggles when a HID packet is received. Therefore, latency here is defined by the interval of the time a command is received from the host device (PC) to the time the Haplet receives the command. We found that with three Haplets receiving commands simultaneously, the median latency is 1.50 ms over 10 s, with a maximum latency of 3.60 ms during

our testing window. A plot of the latency over the time period is shown in **Figure 2B** along with an excerpt of captured packet times with the timeslot correction in use in **Figure 2C**. It should be noted that the test was done in ideal conditions where no packets were lost and the Haplets are in close proximity with the host controller.

2.3 Vibration Amplitude Compensation

Haplet's LRA is an off-the-shelf G1040003D 10 mm LRA module (Jinlong Machinery and Electronics, Inc.). The module has a resonant frequency at 170 Hz and is designed to be used at that frequency, however, since the LRA is placed in such close proximity to the skin, we observed that frequencies as low as 50 Hz at high amplitudes were just as salient as those closer to the resonant frequency at lower amplitudes. Lower frequencies are important for rendering rough textures, pressure, and softness (Kuchenbecker et al., 2006; Choi et al., 2020) and a wide range of frequency is required for rendering realistic textures (Fishel and Loeb, 2012). Thus, we performed simple characterization in order to compensate for the output of the LRA at frequencies outside the resonant frequency range, from 50 to 250 Hz. **Figure 3C** shows the output response of the LRA as supplied by the manufacturer when compared to our own characterization using the characterization jig in **Figure 3A** (as suggested by the Haptics Industry Forum⁴), and when characterized on the fingertips (**Figure 3B**). The acceleration output was recorded using a micro-electromechanical-based inertial measurement unit (MEMS-based IMU) (ICM42688, TDK) on a 6.4 mm by 10.2 mm, 0.8 mm thick PCB connected to a specialized Haplet via an I²C connection through a flat flex ribbon cable (FFC). The specialized Haplet streams readings from the IMU at 1,000 Hz to the host device for recording on a PC. We found that when using the compensation profile derived from our characterization jig (**Figure 3A**) on the fingernail, the output at higher frequencies

⁴High Definition Inertial Vibration Actuator Performance Specification <https://github.com/HapticsIF/HDActuatorSpec>



were severely overcompensated for and provided uncomfortable levels of vibration. However, when characterization was performed at the target site (the fingertips), with the IMU attached on the fingerpad, the resulting output amplitudes after compensation were subjectively pleasant and more consistent to our expectations.

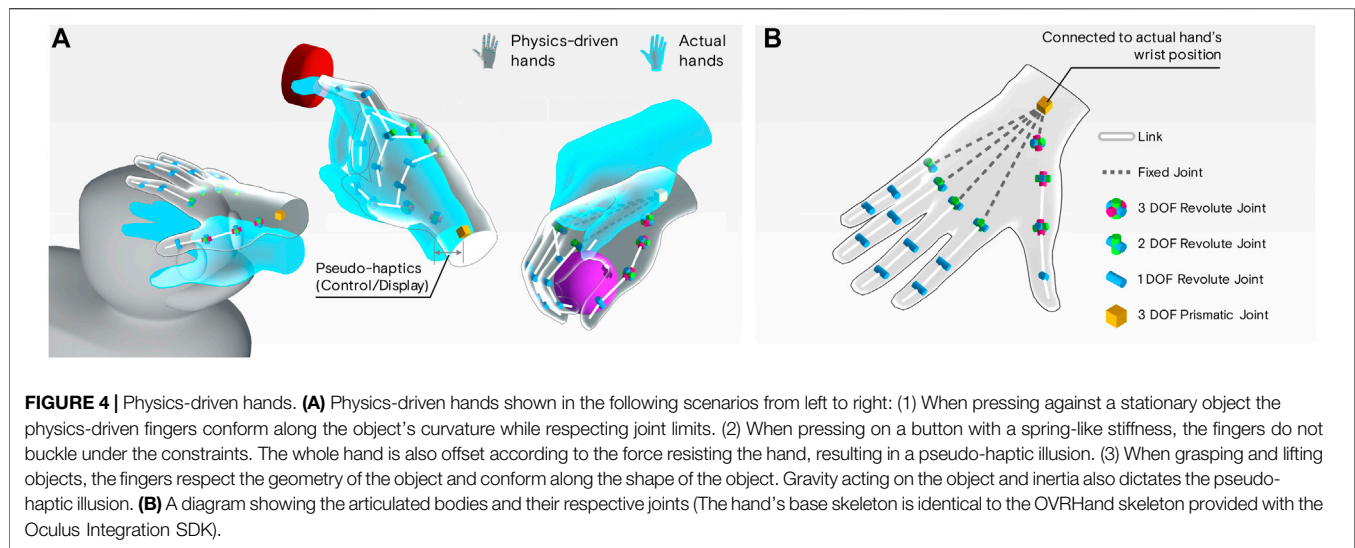
Characterization was performed by rendering sine wave vibrations at frequencies ranging from 50 to 250 Hz in 10 Hz increments and at amplitudes ranging from 0.04 to 1.9 V (peak-to-peak) in 0.2 V increments with a duration of 0.5 s. Acceleration data was sampled and collected at 1,000 Hz during the vibration interval. A total of five repetitions of the frequency and amplitude sweeps were performed. The resulting amplitude is the mean of the maximum measured amplitude of each repetition for each combination of frequency and amplitude, as presented in **Figure 3D**. The output compensation is then calculated first by fitting a linear model for each frequency's response, $amplitude = x_f V$. Then, the inverse of the model is used with the input being the desired acceleration amplitude, in m/s^2 , and the output being the voltage for achieving that acceleration. Frequencies outside the characterized models are linearly interpolated. The results from our compensation on a subset of the characterized frequencies, along with frequencies outside of the characterization intervals, is shown in **Figure 3E**. We use the same compensation profile for every instance of haptic rendering throughout this paper.

2.4 Virtual Environment

Our haptic hardware is only one part of our system. Robust software that can create compelling visuals and audio as well as believable and responsive haptic effects is equally important. In this paper, we build up a software framework using the Unity game engine to create our virtual environments as described in the following sections. Our entire system is run locally on the Oculus Quest two and is completely standalone: requiring only the headset, our USB-connected host microcontroller and the Haplets themselves.

2.4.1 Haptic Rendering

Haplets are commanded to render sine wave vibrations using packets that describe the sine wave frequency, amplitude and duration. Each vibration becomes a building block for haptic effects and are designed to be either used as a single event or chained together for complex effects. For example, a small "click" that resembles a click on a trackpad can be commanded as a 10 millisecond, 170 Hz vibration with an amplitude of $0.2 m/s^2$. To render textures, short pulses of varying frequencies and amplitudes are chained together in rapid succession. Due to the low-latency, haptic effects can be dynamic and responsive to the environment. Examples of interactions that highlight the responsiveness of such a low-latency system are presented in the following sections.



2.4.2 Physics-Driven Hands

The user's hands in our environment are physically simulated using NVIDIA PhysX Articulations system for robotics⁵. Articulations are abstracted as ArticulationBodies in the Unity game engine. This system enables robust hand-object manipulations and believable response towards other rigid bodies in the scene, such as pushing, prodding, and throwing. The fingers are a series of linkages connected using either 1, 2, or 3 degree of freedom revolute joints with joint limits similar to that of a human hand (Cobos et al., 2008). The wrist is connected to the tracked position of actual wrist using a 3 degree of freedom prismatic joint. As a result, pseudo-haptics (Lécuyer, 2009) is readily available as part of the system, meaning that users must extend their limbs further than what is seen in response to a larger force being applied to the virtual hands. This is also known as the pseudo-haptic weight illusion (Samad et al., 2019) or the god object model (Zilles and Salisbury, 1995). For higher fidelity, we set our simulation time step to 5 ms and use the high frequency hand tracking (60 Hz) mode on the Oculus Quest 2. A demonstration of the system is available as a video in the **Supplementary Materials and Figure 4**.

We take advantage of the robust physics simulation and low-latency communication to render haptic effects. A collision event that occurs between a finger and an object is rendered as a short 10 ms burst of vibration with an amplitude scaled to the amount of impulse force. Each object also has unique haptic properties: the frequency of vibration during impact with fingers and the frequency of vibration during fingers sliding across the object. For instance, a wooden surface with high friction would have a sliding frequency of 170 Hz and a rubber-like surface with lower friction would have a sliding frequency of 200 Hz. Additionally, as both our objects and fingers have friction, when a finger glides across a

surface, the stick-slip phenomenon can be observed both visually and through haptic feedback (**Figures 5A,B**).

2.4.3 Tools

With the fingerpads free to grasp and hold actual objects, we augment the presence of handheld tools using visual and haptic feedback.

First, we detect the tool being held in the hand using a technique similar to template matching, as presented in GripMarks (Zhou et al., 2020). Each tool has a unique pose of the hand, such as the pose when holding a pen or the pose when holding a spray bottle (**Figure 6**), which is stored a set of joint angles for every joint of the hand. Our algorithm then compares each tool's predefined pose to the current user's pose using the Pearson correlation coefficient in a sliding 120 frame window. If 80% of frames in the window contains a pose with over a coefficient over 0.9, then it is deemed that the tool is being held in the user's hand. To "release" a tool, the user would simply open their hand fully for 1 s (**Figure 6**). Any tool can now be altered in shape and experience both visually and through haptics through the headset. For instance, the user can physically hold a pen but in a pose akin to holding a hammer, and in their VR environment, they would see and feel as if they are holding a hammer. Additionally, since our algorithm relies only on the hand's pose, an actual tool does not have to be physically held by the user's hand, we also explore this in our user study in the following sections.

When a tool is detected, the tool is visually rendered attached to the hand. In our physics simulation, the tool's rigid body is attached to the wrist's ArticulationBody and thus the tool can respond dynamically to the environment as if the tool and the hands are a single object, maintaining the same pseudo-haptic capabilities as presented in the previous sections. The physically held tools provide passive haptic feedback in the form of pressure and familiar grounding while Haplets can be used to render vibrotactile feedback to augment the presence of the held tool in response to the virtual environment. Three examples of haptic rendering schemes for tools are presented in **Figure 7**.

⁵<https://gameworksdocs.nvidia.com/PhysX/4.0/documentation/PhysXGuide/Manual/Articulations.html>

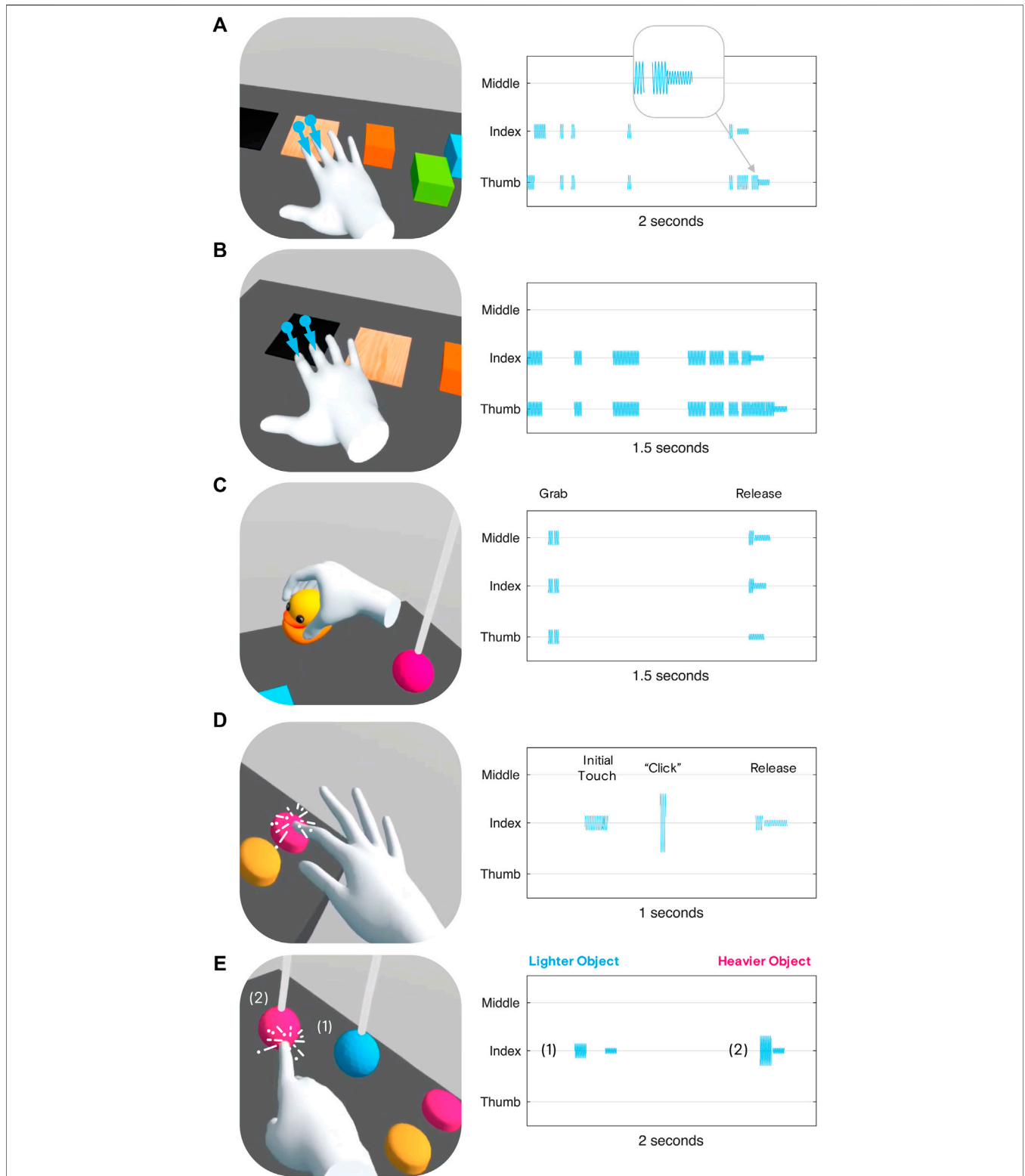
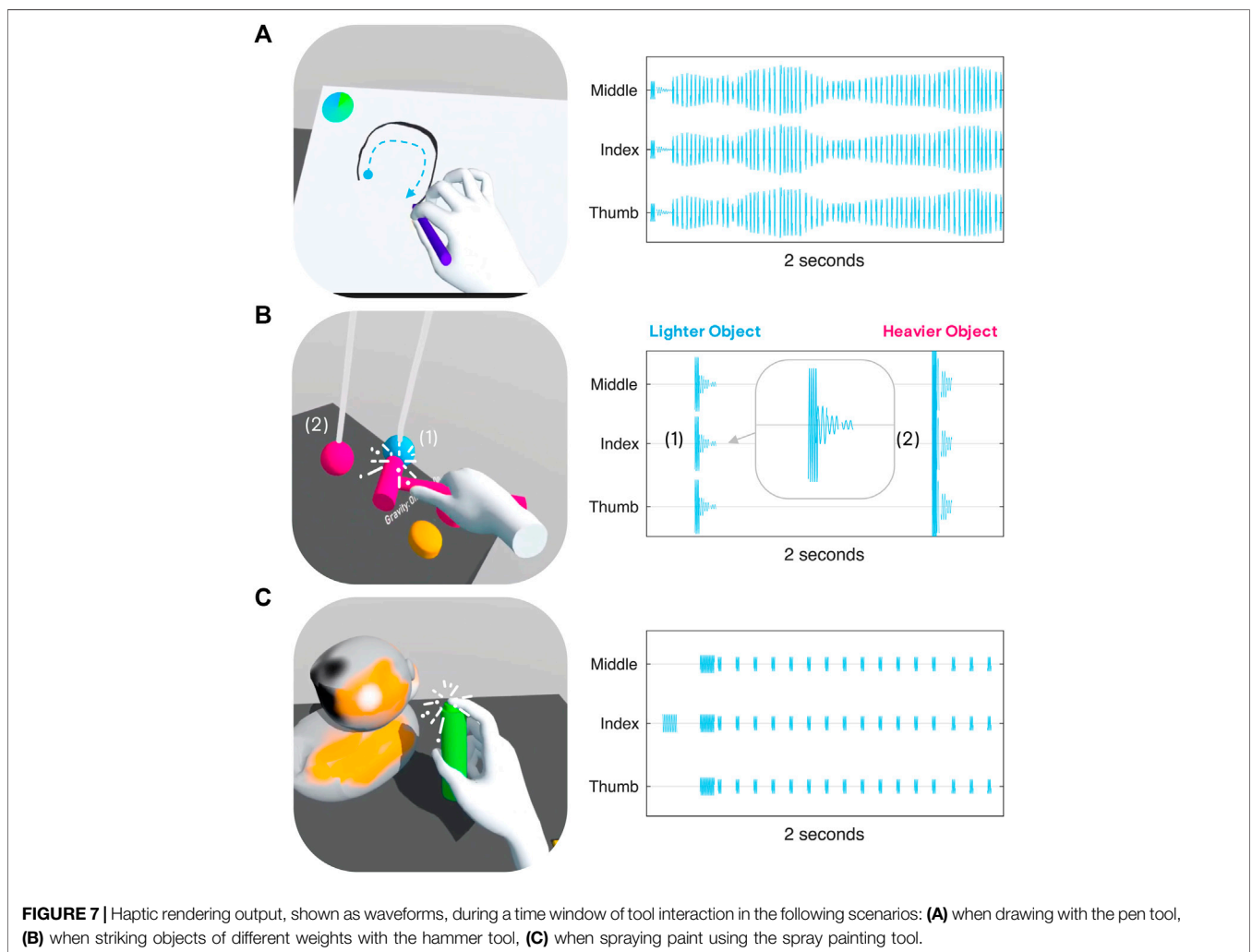
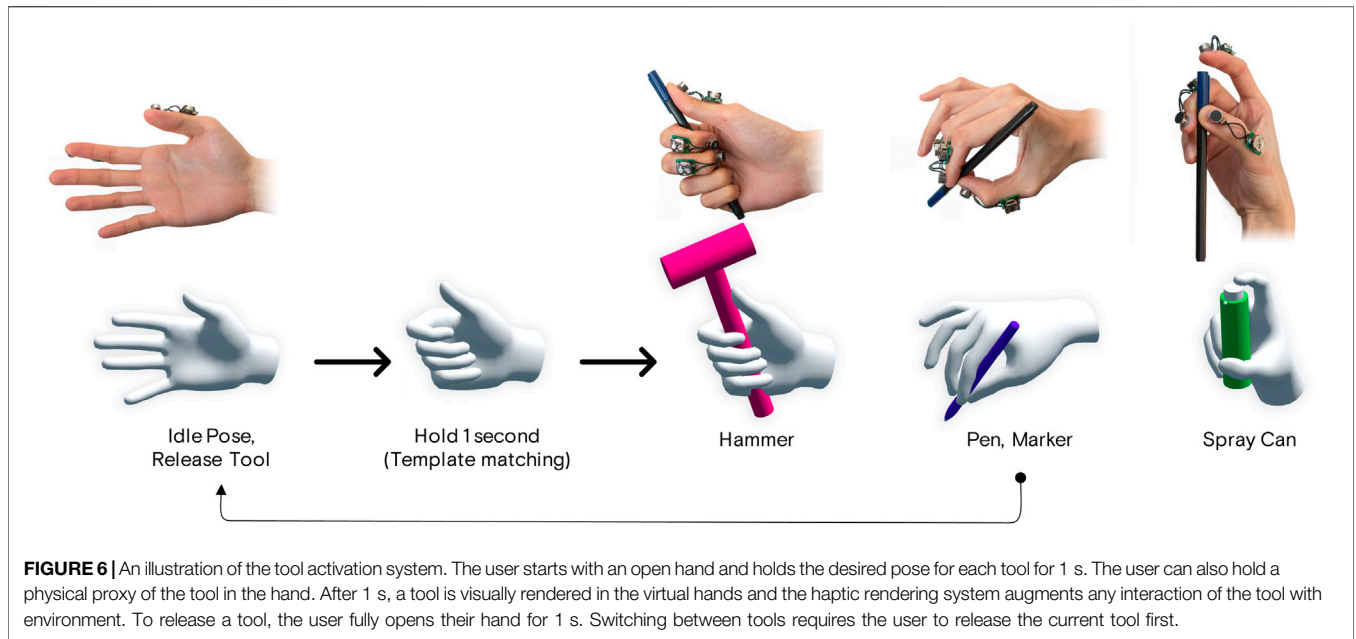


FIGURE 5 | Haptic rendering output, shown as waveforms, during a time window of interaction in the following scenarios: **(A)** dragging fingers across wood, **(B)** dragging fingers across smooth plastic, **(C)** picking up and letting go of a plastic toy, **(D)** pressing a button, and **(E)** prodding two similar spheres with different masses.



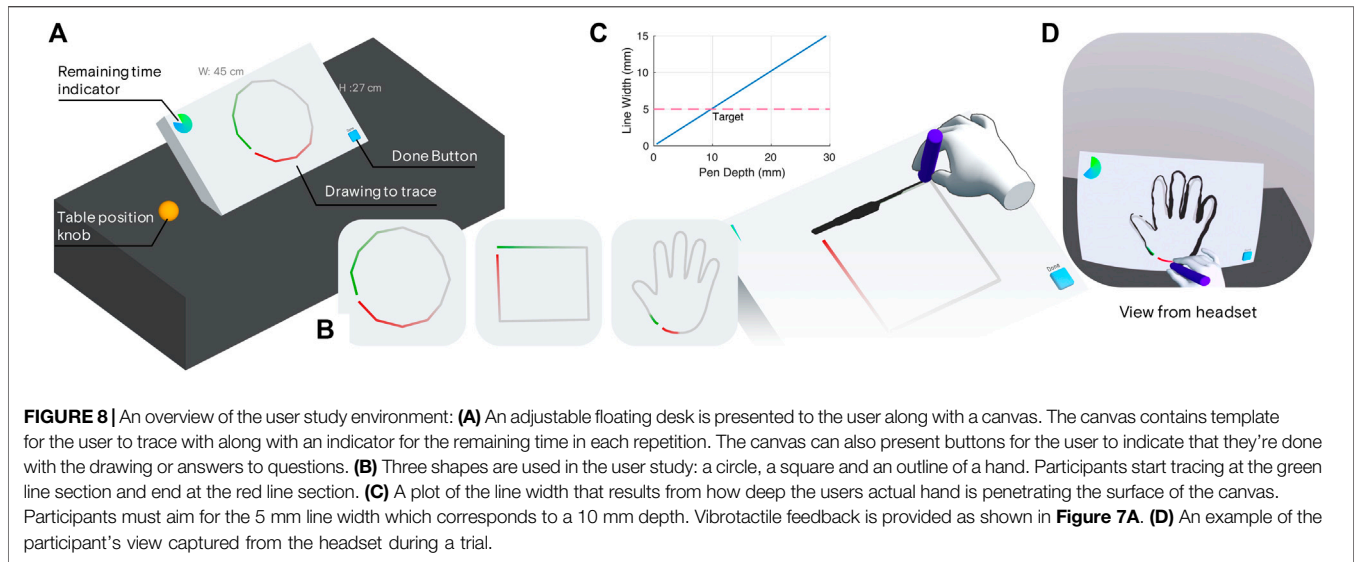


FIGURE 8 | An overview of the user study environment: **(A)** An adjustable floating desk is presented to the user along with a canvas. The canvas contains template for the user to trace with along with an indicator for the remaining time in each repetition. The canvas can also present buttons for the user to indicate that they're done with the drawing or answers to questions. **(B)** Three shapes are used in the user study: a circle, a square and an outline of a hand. Participants start tracing at the green line section and end at the red line section. **(C)** A plot of the line width that results from how deep the users actual hand is penetrating the surface of the canvas. Participants must aim for the 5 mm line width which corresponds to a 10 mm depth. Vibrotactile feedback is provided as shown in **Figure 7A**. **(D)** An example of the participant's view captured from the headset during a trial.

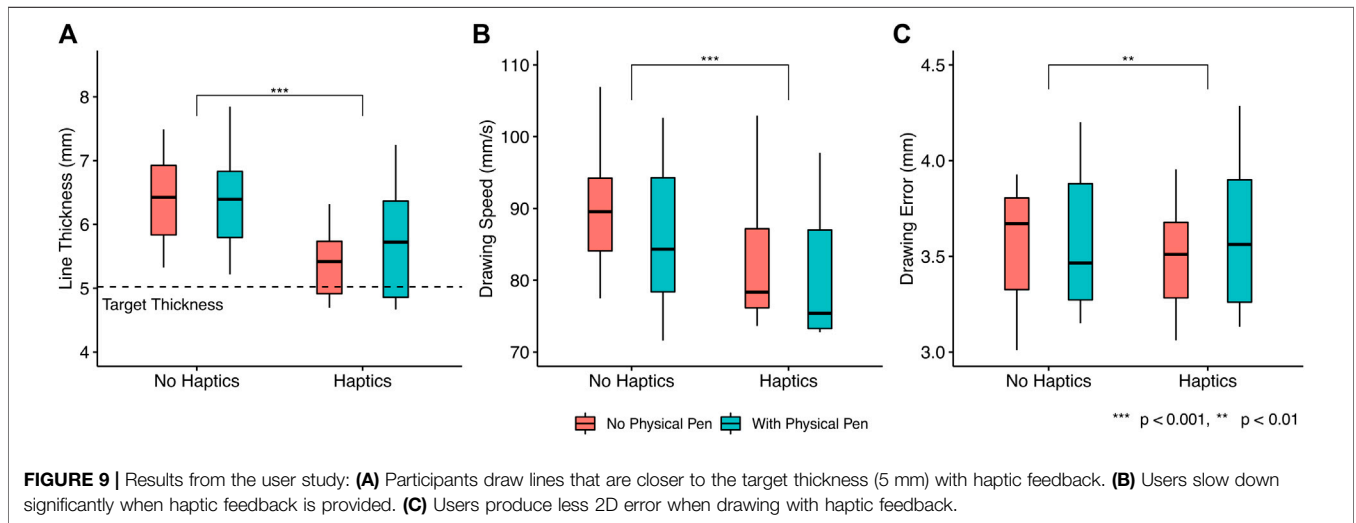


FIGURE 9 | Results from the user study: **(A)** Participants draw lines that are closer to the target thickness (5 mm) with haptic feedback. **(B)** Users slow down significantly when haptic feedback is provided. **(C)** Users produce less 2D error when drawing with haptic feedback.

3 USER STUDY

In this section, we introduce a sketching user study to evaluate the feasibility of using Haplets in a productivity scenario. We chose a sketching task because it encompasses the main concepts introduced in this paper: 1) a physical tool (a pen) is held by the user, thus allowing Haplets to augment the tool with vibrotactile haptics, 2) upon the pen contacting with a surface and while drawing, Haplets renders impacts and textures, and 3) physics-driven hands and tools respond to the sketching environment, introducing pseudo-haptic force and friction. The main task is loosely based on VRsketchPen (Elsayed et al., 2020) where the user would trace a shape shown on a flat canvas. With a simulated pressure sensitive pen, users would need to maintain a precise distance from the canvas in order to draw a line that matches the line thickness of the provided guide. We hypothesize that with Haplets providing vibrotactile

feedback, users would be able to draw lines closer to the target thickness. In addition to the sketching task, after the end of the session, the user is presented with a manipulation sandbox for them to explore the remaining modalities that Haplets has to offer such as texture discrimination, object manipulation, and pseudo-haptic weight. The details for this sandbox is described in **Section 4.1**.

3.1 Experimental Setup

We recruited eight right-handed participants (2 females, aged 22–46, mean = 32.75, SD = 7.44) to participate in the study. Proper social distancing and proactive disinfection according to local guidance was maintained at all times and the study was mostly self-guided through prompts in the VR environment. The study was approved by our institution's IRB and participants gave informed consent. Participants were first seated and started by donning three Haplets on the thumb, index and middle fingers of

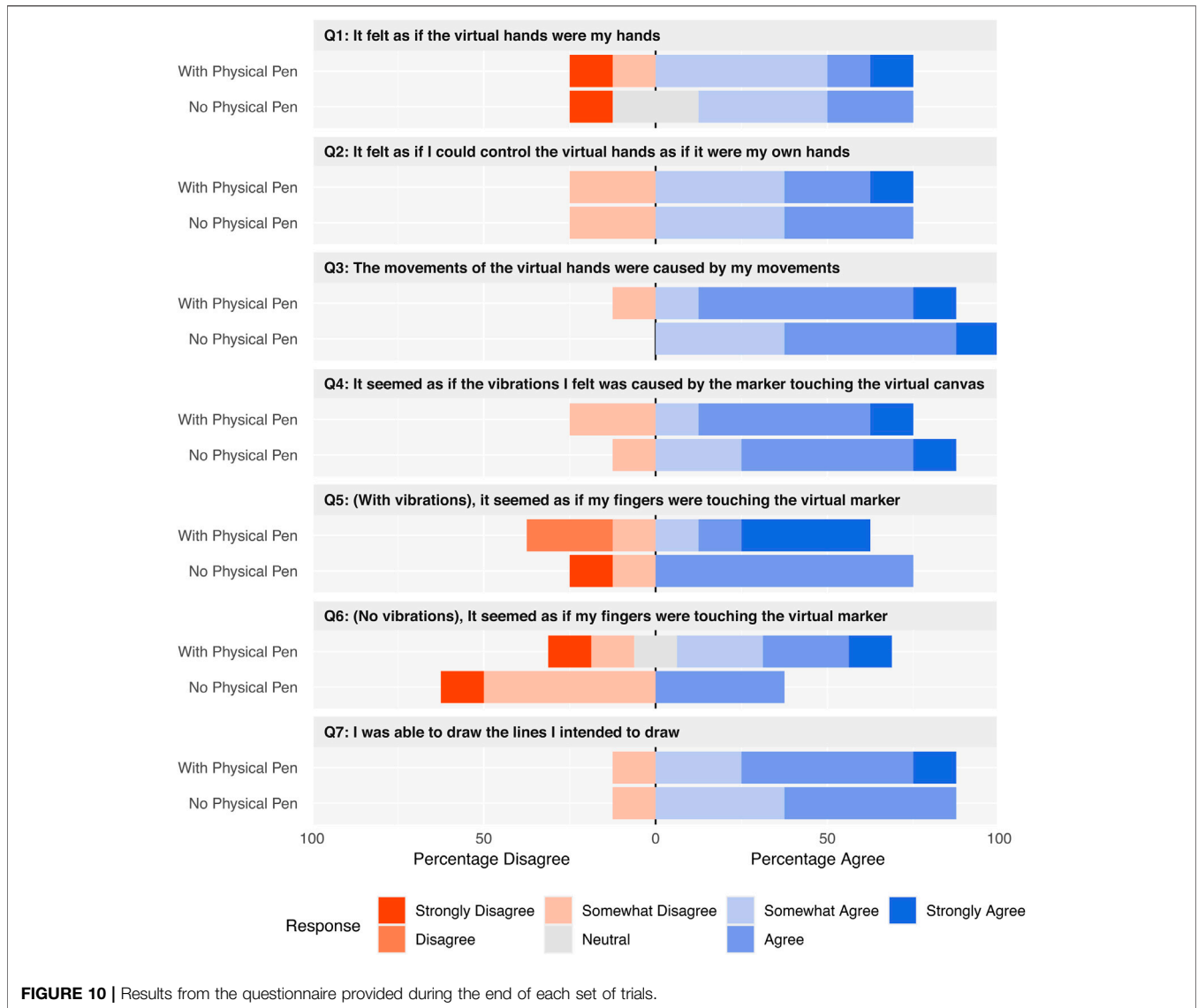


FIGURE 10 | Results from the questionnaire provided during the end of each set of trials.

their right hand. Then they donned an Oculus Quest two headset. Participants then picked up a physical pen and held it in their left hand using the headset’s AR Passthrough mode, then a standalone VR Unity application was launched. All experiments were run and logged locally on the headset. All interactions in VR were done using on-device hand tracking (Han et al., 2020).

The VR environment consists of a single desktop with a large canvas, as shown in **Figure 8**. The canvas is used to present instructions, questions and the actual tracing task. Participants were first instructed to hold the physical pen in their right hand, which will also create a virtual pen in their hand using the tool detection system presented in the previous section. The participant uses the pen to interact with most elements in the environment including pressing buttons and drawing.

The main task consists of participants tracing three shapes with the virtual pen: a square, a circle and a hand, as shown in **Figure 8**. The shapes are presented in a randomized order and

each shape is given 15 s to complete. The virtual pen is pressure sensitive and the lines the users draw vary in thickness depending on how hard the user is pressing against the canvas—we simulate this using our physically simulated hands, which means that the further the user’s real hands interpenetrates the canvas, the thicker the line will be. The target thickness for all shapes is 5 mm. Lines are rendered in VR using the Shapes real-time vector library⁶.

When drawing, for every 2.5 mm the pen has traveled on the canvas, Haplets render a vibration for 10 ms at 170 Hz with an amplitude that is mapped to how much virtual force is exerted on the canvas. Since the amount of force is also proportional to the line width, the amplitude of vibration is also mapped to the line width, as shown in **Figure 8C**. In other words, the pen and Haplets emulates the sensation of drawing on a rough surface and

⁶Shapes by Freya Holmér <https://acegikmo.com/shapes>

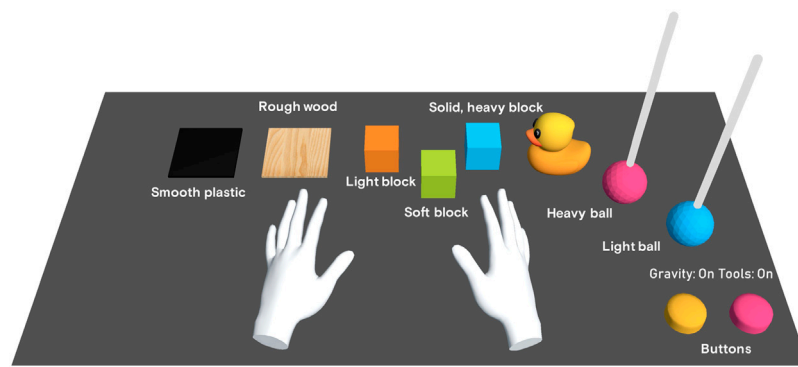


FIGURE 11 | After the user study concludes, participants are presented with a desk with various objects to interact with. From left to right: a smooth plastic square, a rough wooden square, a lightweight orange block, a soft green block, a heavy blue block, a rubber ducky (solid), a heavy tethered ball, a lightweight tethered ball. Participants can use the lower right buttons to toggle the use of tools and toggle gravity on and off. Each object responds with haptic feedback as shown in **Figures 5, 7**.

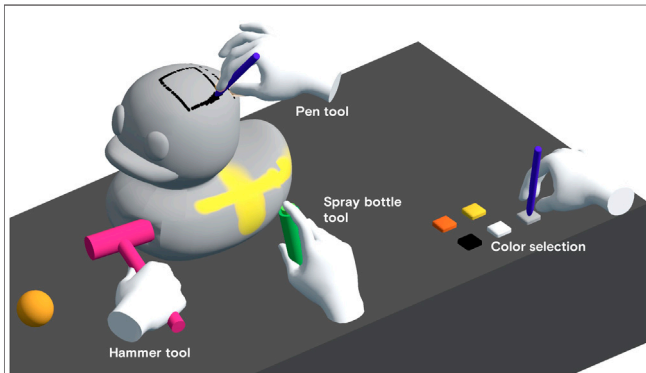


FIGURE 12 | The painting application highlights the use of haptic feedback to enhance the experience of using otherwise passive tools in the hand. Shown are four interactions overlaid from left to right: using the hammer tool to adjust the sculpture's geometry, using the pen tool to draw on the sculpture, using the spray bottle to spray paint on the sculpture, and selecting colors by tapping the tools on the swatches provided. Haptic feedback provided by the tools is visualized in **Figure 7**.

the pressure is represented as the strength of vibration. An example of the haptic rendering scheme's output is shown in **Figure 7A**.

Participants first perform 12 trials in a training phase to get familiar with the task where they would hold a physical pen but would not receive haptic feedback. Participants then performed two sets of 36 trials (12 of each shape per set) with or without holding the physical pen in their hand, totaling 72 trials. We balance the order of this throughout our participants to account for any order effects. Half of the 36 trials in each set have the Haplets turned off, presented in a randomized order. In summary, participants are given two conditions for either holding or not holding the pen and two conditions for either having or not having haptic feedback. After each set of trials, they

are presented with a short questionnaire. After all trials have concluded, the participant is given a short demo of other capabilities of Haplets, as described in the following sections.

For each trial, we collected the line thickness for every line segment, the coordinates along each line segment, and the time it took to complete the drawing. Through post-processing, we calculate the mean line thickness for each trial, the mean drawing speed and the mean 2D error from the given guide.

3.2 Results

We performed a two-way repeated measures analysis of variance (ANOVA) for each independent variable: line thickness, drawing speed and drawing error with two within-subject factors: with or without haptic feedback, and with or without a physical pen held. Our analysis, as presented in **Figure 9**, revealed main effects for haptic feedback for line thickness ($F(1,7) = 15.82, p < 0.01$), drawing speed ($F(1,7) = 10.75, p < 0.02$), and drawing error ($F(1,7) = 14.24, p < 0.01$), but non-significance for the physical pen conditions nor the interactions between factors. Pairwise *t*-tests between the haptic and non-haptic conditions for each independent variable confirmed significance in line thickness ($p < 0.001$), drawing speed ($p < 0.001$), and drawing error ($p = 0.001$).

For line thickness, we can observe that participants can rely on the haptic feedback for guidance and draw lines that are closer to the guide's thickness, with an mean error across subjects of 1.58 mm with haptic feedback and 2.68 mm without haptic feedback. Having a physical pen in the hand seems to negatively impact the line thickness, we attribute this to the deteriorated tracking accuracy when the physical pen occludes parts of the tracked fingers—a limitation of our particular setup. We confirmed this by observing video recordings of the sessions, where we could correspond moments of large line width variations to temporary losses of hand tracking (indicated by malformed hand rendering or hand disappearance). We can also observe that participants slow down significantly when haptic feedback is provided. We hypothesize that participants were

actively using the haptic feedback to guide their strokes. The reduced drawing error is most likely to be influenced directly by the reduced speed of drawing.

When interviewed during a debriefing session after the experiment, two participants (P3 and P5) noted that they “forgot (the Haplets) were there”. Some participants (P1, P3 and P8) preferred having the physical pen in their hands, while other participants (P2 and P7) did not. One participant (P7) suggests that they were more used to using smaller styli and thus preferred not having the larger pen when using a virtual canvas. Another participant (P6) noted that they felt the presence of the physical pen even after it has been removed from their hands.

From our questionnaire (**Figure 10**), we can observe that participants had a reasonable amount of body ownership (Q1) and agency (Q2, Q3). Having a physical pen in their hands did not seem to alter the experience of drawing on a virtual canvas (Q4, Q7). However, the presence of holding a virtual pen in VR seems to be positively impacted by having active haptic rendering (from the Haplets) along with the passive haptic feedback from holding a physical pen (Q5, Q6).

4 DEMONSTRATION

We built demonstrations for highlighting two potential use-cases for Haplets: 1. manipulation in AR/VR and 2. a painting application that uses our tool system.

4.1 Manipulation Sandbox

The manipulation sandbox is a demonstration presented to participants at the end of our user study. The user is presented with a desk with several objects and widgets, as shown in **Figure 11**. A video showing the demonstration is provided with the **Supplementary Materials**.

The top-left corner of the desk consists of two squares with two different textures. The left square represents a smooth, black, plastic surface and the right square represents a grainy, wooden surface. The smooth surface has a low coefficient of friction of 0.1 while the rough surface has a high coefficient of friction of 1.0. When the user runs their fingers across each surface, the haptics system renders different frequencies for each texture, at 200 and 100 Hz, respectively. Each vibration is generated after the fingertips have traveled at least 2.5 mm, similarly to the pen’s haptic rendering scheme presented in the previous section. Since the wooden texture has higher friction, the user’s finger will stick and slip, rendering both visually and through haptics, the sensation of a rough surface. An example of the surfaces’ haptic rendering system in use is shown in **Figures 5A,B**.

Towards the center of the desk are three cubes with varying densities. The user can either pick up or prod the cubes to figure out which cube is lighter or heavier than the others. When prodded at, visually, the lighter cube will slide while and heavier cube will topple. When picked up, the lighter cube will render a lower control-display ratio (pseudo-haptic weight illusion) than the heavier cubes. Each cube also responds to touch differently. The lighter cube will render a vibration of

100 Hz upon touch, to simulate a softer texture, while the heavier cube will render a vibration of 200 Hz to simulate contact with a dense object. Held cubes can also be tapped against the desk, and similar vibrations will be rendered upon impact. Upon release, Haplets will render a smaller amplitude vibration of the same frequency. A rubber duck is also presented nearby, with similar properties to the cubes. An example of the haptic rendering output is shown in **Figure 5C**.

Towards the right of the desk are two buttons: one for toggling gravity on and off and another for toggling the tool system on and off. The button responds to initial touch using the same system as other objects but emit a sharp click (170 Hz, 20 ms) when depressed a certain amount to signal that the button has activated. If the user turns the tool system off, tools will not be created when a pose is recognized. The buttons’ haptic rendering scheme is shown in **Figure 5D**.

When the tool system is active, users can create a hammer in their hand by holding the “thumbs-up” pose (see **Figure 6**). The hammer can be used to tap and knock the items on the desk around. The hammer’s haptic system is similar to the fingertips, where each object responds with different frequencies depending on pre-set properties and different amplitudes depending on how much reaction force is generated when the hammer strikes the object. All three Haplets will vibrate upon hammer strikes under the assumption that the user is holding the hammer with all three fingers. Additionally, a lower frequency reverberation is also rendered immediately after the initial impact in order to emulate stiffness. An example of the haptic rendering scheme for the hammer is shown in **Figure 7B**.

4.2 Painting

Our painting demonstration, as shown in **Figure 12**, is designed to highlight the use of our tool system. A desk with a large, gray duck sculpture is presented to the user. The sculpture can be rotated using the user’s bare hands. The user can use three tools during the demonstration: a spray bottle, a pen and a hammer. Each tool is placed in the user’s hand when they produce the correct pose, as shown in **Figure 6**. The lower right corner of the desk contains a palette of five colors: the user can tap the tool on the color to switch the tool to operate with that color. A video showing the demonstration is provided with the **Supplementary Materials**.

The spray bottle is used to quickly paint the duck sculpture. As the user presses down on the bottle’s nozzle, a small click is rendered on the index finger Haplet. When the nozzle is engaged and the tool is producing paint, all three Haplets pulse periodically with an amplitude that corresponds to how much the nozzle is depressed. An example of the haptic rendering output is shown in **Figure 7C**. Paint is deposited onto the sculpture similar in behavior to spray painting.

The pen is used to mark fine lines on the sculpture. The haptic rendering scheme is identical to that of the pen described in the user study, where all three Haplets render vibrations with amplitudes that correspond to the depth of penetration and line width.

The hammer is used to modify the sculpture by creating indentations. The sculpture’s mesh is modified in response to

the reaction force caused by strikes of the hammer. Haptic feedback for the hammer is similar to that presented in the previous section (**Figure 7B**).

5 DISCUSSION

Haplets introduces a wireless, finger-mounted haptic display for AR and VR that leaves the user's fingertip free to interact with real-world objects, while providing responsive vibrotactile haptic rendering. Each Haplet is a self-contained unit with a footprint small enough to fit on the back of the fingers and fingernail. We also present an engineering solution to achieve low-latency wireless communication that adds haptic rendering to various use cases in VR such as manipulation, texture rendering, and tool usage. Our simulation system for physics-driven virtual hands complements our haptic rendering system by providing pseudo-haptics, robust manipulation, and realistic friction. With a real-world tool held in the hand, Haplets render vibrotactile feedback along with visuals from our simulation system to augment the presence of the tool. Our user study and demonstrations show that Haplets is a feasible solution for a low-encumbrance haptic device.

Although Haplets exclusively provide vibrotactile feedback, we have introduced several engineering efforts to maximize the rendering capabilities of our haptic actuator, the LRA. Our brief characterization of the LRA shows that LRAs can be used at frequencies outside the resonant frequency when properly compensated for. Furthermore, our characterization also shows that the material (or body part) on which the actuator is mounted on to can cause the output of the actuator to vary significantly and therefore needs to be characterized for the intended location of the actuator. Our low-latency wireless communication also helps minimize the total latency from visual stimuli to tactile stimuli, which is especially useful when considering the inherent mechanical time delay for LRAs.

Our user study and subjective feedback from the demonstrations have shown that Haplets and the current framework do provide adequate haptic feedback for the given tasks and experiences. However, the human hand can sense much more than simple vibrations such as the sensation of shear force, normal force, and temperature. We address this shortcoming by introducing believable visuals in the form of physics-driven hands, and make up for the lack of force rendering by introducing passive haptic feedback in the form of tools. Our low-latency solution enables impacts, touch and textures to be rendered responsively according to the simulation and visuals. Furthermore, we have yet to fully explore the voice coil-like rendering capabilities of the haptic actuator (LRA). Therefore, our immediate future improvement to our system is the ability to directly stream waveform data to the device. This will enable the ability to render arbitrary waveforms on the LRAs or VCMs which can be used to render highly realistic textures (Chan et al., 2021) or the use of audio-based tools for authoring haptic effects (Israr et al., 2019; Pezent et al., 2021).

Our implementation of passive haptic feedback for tools uses a pen for physical grounding of the fingers, which

provides adequate grounding for a number of tasks. Inertial cues that provide the sense of weight to the pen are presented using pseudo-haptic weight. Shigeyama et al. (2019) have shown that VR controllers with reconfigurable shapes can provide realistic haptic cues for inertia and grounding. Therefore, a potential venue for future work is the use of Haplets in conjunction with actual tools (e.g., holding an actual hammer or an actual spray can) or reconfigurable controllers, which would not only provide realistic grips and inertia but also additional haptic feedback that the tool may provide such as depressing the nozzle of a spray bottle.

For other potential future work, our framework provides a foundation for building wearable haptic devices which are not necessarily limited to the fingers. In its current form, Haplets can be placed on other parts of the body with minimal adjustments for rapid prototyping haptic devices, such as the forearm, temple, and thighs (Cipriani et al., 2012; Sie et al., 2018; Rokhmanova and Rombokas, 2019; Peng et al., 2020). With some modifications, namely to the number of motor drivers and firmware, Haplets can also be used for rendering a larger number of vibrotactile actuators at once, which could be used to create haptic displays around the wrist or on the forearm. Furthermore, our motor drivers are not limited to driving vibrotactile actuators, skin stretch and normal force can be rendered with additional hardware and DC motors (Preechayasomboon et al., 2020).

6 CONCLUSION

We have introduced Haplets as a wearable haptic device for fingers in VR that is low encumbrance. Haplets can augment the presence of virtual hands in VR and we strengthen that further with physics-driven hands that respond to the virtual environment. We have also introduced an engineering solution for achieving low-latency wireless haptic rendering. Our user study and demonstration shows that Haplets have potential in improving hand and tool-based VR experiences. Our system as a whole provides a framework for prototyping haptic experiences in AR and VR and our immediate future work is exploring more use cases for Haplets.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding authors.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by University of Washington IRB. The participants

provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

Both authors conceptualized the device. PP designed and built the hardware, firmware and software for the devices, user study and demonstrations. Both authors conceptualized and designed the user study. PP ran the user study. PP wrote the first draft of the manuscript. Both authors contributed to manuscript revision, read, and approved the submitted version.

REFERENCES

- Achibet, M., Le Gouis, B., Marchal, M., Léziart, P.-A., Argelaguet, F., Girard, A., Lecuyer, A., and Kajimoto, H. (2017). "FlexiFingers: Multi-finger Interaction in VR Combining Passive Haptics and Pseudo-haptics," in 2017 IEEE Symposium on 3D User Interfaces (3DUI), 18-19 March 2017, Los Angeles, CA, USA, 103–106. doi:10.1109/3DUI.2017.7893325
- Ando, H., Kusachi, E., and Watanabe, J. (2007). "Nail-mounted Tactile Display for Boundary/texture Augmentation," in Proceedings of the international conference on Advances in computer entertainment technology, 13-15 June 2007, Salzburg, Austria (New York, NY, USA: Association for Computing Machinery), 292–293. ACE '07. doi:10.1145/1255047.1255131
- Caballero, D. E., and Rombokas, E. (2019). "Sensitivity to Conflict between Visual Touch and Tactile Touch," in IEEE Transactions on Haptics 12, 78–86. doi:10.1109/TOH.2018.2859940
- Chan, S., Tymms, C., and Colonnese, N. (2021). *Hasti: Haptic and Audio Synthesis for Texture Interactions*, 6.
- Choi, I., Zhao, E., Gonzalez, E. J., and Follmer, S. (2020). "Augmenting Perceived Softness of Haptic Proxy Objects through Transient Vibration and Visuo-Haptic Illusion in Virtual Reality," in IEEE Transactions on Visualization and Computer Graphics, 1. doi:10.1109/TVCG.2020.3002245
- Cipriani, C., D'Alonzo, M., and Carrozza, M. C. (2012). "A Miniature Vibrotactile Sensory Substitution Device for Multifingered Hand Prosthetics," in IEEE Transactions on Biomedical Engineering 59, 400–408. doi:10.1109/TBME.2011.2173342
- Cobos, S., Ferre, M., Sanchez Uran, M., Ortego, J., and Pena, C. (2008). "Efficient Human Hand Kinematics for Manipulation Tasks," in 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, 22-26 September 2008, Nice, France, 2246–2251. doi:10.1109/IROS.2008.4651053
- Elsayed, H., Barrera Machuca, M. D., Schaarschmidt, C., Marky, K., Müller, F., Riemann, J., Matvienko, A., Schmitz, M., Weigel, M., and Mühlhäuser, M. (2020). "VRSketchPen: Unconstrained Haptic Assistance for Sketching in Virtual 3D Environments," in 26th ACM Symposium on Virtual Reality Software and Technology, 1-4 November 2020, Virtual Event, Canada (New York, NY, USA: Association for Computing Machinery), 1–11. VRST '20. doi:10.1145/3385956.3418953
- Fishel, J. A., and Loeb, G. E. (2012). Bayesian Exploration for Intelligent Identification of Textures. *Front. Neurobot.* 6. doi:10.3389/fnbot.2012.00004
- Gupta, A., Samad, M., Kin, K., Kristensson, P. O., and Benko, H. (2020). "Investigating Remote Tactile Feedback for Mid-air Text-Entry in Virtual Reality," in 2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 9-13 November 2020, Porto de Galinhas, Brazil, 350–360. doi:10.1109/ISMAR50242.2020.00062
- Han, S., Liu, B., Cabezas, R., Twigg, C. D., Zhang, P., Petkau, J., et al. (2020). MEgATrack. *ACM Trans. Graph.* 39. doi:10.1145/3386569.3392452
- Hinche, R., Vechev, V., Shea, H., and Hilliges, O. (2018). "DextrES," in Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology, 14-17 October 2018, Berlin, Germany (New York, NY, USA: Association for Computing Machinery), 901–912. UIST '18. doi:10.1145/3242587.3242657
- Hsieh, M.-J., Liang, R.-H., and Chen, B.-Y. (2016). "NailFactors," in Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services, 6-9 September 2016, Florence, Italy. (New York, NY, USA: Association for Computing Machinery), 29–34. MobileHCI '16. doi:10.1145/2935334.2935358
- Israr, A., Zhao, S., Schwemler, Z., and Fritz, A. (2019). "Stereohaptics Toolkit for Dynamic Tactile Experiences," in *HCI International 2019 - Late Breaking Papers*. Editor C. Stephanidis (Cham: Springer International Publishing), 217–232. Lecture Notes in Computer Science. doi:10.1007/978-3-030-30033-3_17
- Johansson, R. S., and Flanagan, J. R. (2009). Coding and Use of Tactile Signals from the Fingertips in Object Manipulation Tasks. *Nat. Rev. Neurosci.* 10, 345–359. doi:10.1038/nrn2621
- Kim, H., Yi, H., Lee, H., and Lee, W. (2018). HapCube. doi:10.1145/3173574.317407513
- Kuchenbecker, K. J., Fiene, J., and Niemeyer, G. (2006). "Improving Contact Realism through Event-Based Haptic Feedback," in IEEE Transactions on Visualization and Computer Graphics 12, 219–230. doi:10.1109/TVCG.2006.32
- Lécuyer, A. (2009). Simulating Haptic Feedback Using Vision: A Survey of Research and Applications of Pseudo-haptic Feedback. *Presence: Teleoperators and Virtual Environments* 18, 39–53. doi:10.1162/pres.18.1.39
- Lee, J., Sinclair, M., Gonzalez-Franco, M., Ofek, E., and Holz, C. (2019). "TORC: A Virtual Reality Controller for In-Hand High-Dexterity Finger Interaction," in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. 6-9 May 2019, Glasgow, Scotland UK. (New York, NY, USA: Association for Computing Machinery), 1–13. doi:10.1145/3290605.3300301
- Lengyel, G., Zalalytė, G., Pantelides, A., Ingram, J. N., Fiser, J., Lengyel, M., et al. (2019). Unimodal Statistical Learning Produces Multimodal Object-like Representations. *eLife* 8, e43942. doi:10.7554/eLife.43942
- Peng, Y.-H., Yu, C., Liu, S.-H., Wang, C.-W., Taelle, P., Yu, N.-H., and Chen, M. Y. (2020). "WalkingVibe: Reducing Virtual Reality Sickness and Improving Realism while Walking in VR Using Unobtrusive Head-Mounted Vibrotactile Feedback," in Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 25-30 April 2020, Honolulu, HI, USA (New York, NY, USA: Association for Computing Machinery), 1–12. CHI '20. doi:10.1145/3313831.3376847
- Pezent, E., Cambio, B., and OrMalley, M. K. (2021). "Syntacts: Open-Source Software and Hardware for Audio-Controlled Haptics," in IEEE Transactions on Haptics, 14, 225–233. doi:10.1109/TOH.2020.3002696
- Pezent, E., Israr, A., Samad, M., Robinson, S., Agarwal, P., Benko, H., and Colonnese, N. (2019). "Tasbi: Multisensory Squeeze and Vibrotactile Wrist Haptics for Augmented and Virtual Reality," in 2019 IEEE World Haptics Conference (WHC), 9-12 July 2019, Tokyo, Japan, 1–6. doi:10.1109/WHC.2019.8816098
- Preechayasomboon, P., Israr, A., and Samad, M. (2020). "Chasm: A Screw Based Expressive Compact Haptic Actuator," in Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 25-30 April 2020, Honolulu, HI, USA, (New York, NY, USA: Association for Computing Machinery), 1–13. CHI '20. doi:10.1145/3313831.3376512
- Rekimoto, J. (2009). "SenseableRays," in Proceedings of the 27th international conference extended abstracts on Human factors in computing systems - CHI EA '09 (Boston, MA, USA, 4-9 April 2009, Boston, MA, USA : ACM Press), 2519. doi:10.1145/1520340.1520356

ACKNOWLEDGMENTS

The authors would like to thank current and former members of Rombolabs for their valuable input and insightful discussions: David Boe, Maxim Karrenbach, Abhishek Sharma, and Astrini Sie.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/frvir.2021.738613/full#supplementary-material>

- Rokhmanova, N., and Rombokas, E. (2019). "Vibrotactile Feedback Improves Foot Placement Perception on Stairs for Lower-Limb Prosthesis Users," in 2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR), 24-28 June 2019, Toronto, ON, Canada, 1215-1220. doi:10.1109/ICORR.2019.8779518
- Salazar, S. V., Pacchierotti, C., de Tinguy, X., Maciel, A., and Marchal, M. (2020). "Altering the Stiffness, Friction, and Shape Perception of Tangible Objects in Virtual Reality Using Wearable Haptics," in IEEE Transactions on Haptics 13, 167-174. doi:10.1109/TOH.2020.2967389
- Samad, M., Gatti, E., Hermes, A., Benko, H., and Parise, C. (2019). "Pseudo-Haptic Weight," in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, 4-9 May 2019, Glasgow, Scotland UK. (Glasgow Scotland UK: ACM), 1-13. doi:10.1145/3290605.3300550
- Schorr, S. B., and Okamura, A. M. (2017). "Fingertip Tactile Devices for Virtual Object Manipulation and Exploration," in Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, 6-11 May 2017, Denver, CO, USA. (New York, NY, USA: Association for Computing Machinery), 3115-3119. CHI '17. doi:10.1145/3025453.3025744
- Shigeyama, J., Hashimoto, T., Yoshida, S., Narumi, T., Tanikawa, T., and Hirose, M. (2019). "Transcalibur: A Weight Shifting Virtual Reality Controller for 2D Shape Rendering Based on Computational Perception Model," in Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, 4-9 May 2019, Glasgow, Scotland UK. (New York, NY, USA: Association for Computing Machinery), 1-11. doi:10.1145/3290605.3300241
- Sie, A., Boe, D., and Rombokas, E. (2018). "Design and Evaluation of a Wearable Haptic Feedback System for Lower Limb Prostheses during Stair Descent," in 2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob), 26-29 August 2018, Enschede, Netherlands, 219-224. doi:10.1109/BIOROB.2018.8487652
- Teng, S.-Y., Li, P., Nith, R., Fonseca, J., and Lopes, P. (2021). "Touch&Fold: A Foldable Haptic Actuator for Rendering Touch in Mixed Reality," in Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, 8-13 May 2021, Yokohama, Japan. (New York, NY, USA: Association for Computing Machinery). 1-14. doi:10.1145/3411764.3445099
- Zhou, Q., Sykes, S., Fels, S., and Kin, K. (2020). "Gripmarks: Using Hand Grips to Transform In-Hand Objects into Mixed Reality Input," in Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 25-30 April 2020, Honolulu, HI, USA, (New York, NY, USA: Association for Computing Machinery), 1-11. CHI '20. doi:10.1145/3313831.3376313
- Zilles, C. B., and Salisbury, J. K. (1995). "A Constraint-Based God-Object Method for Haptic Display," in Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots, 5-9 August 1995, Pittsburgh, PA, USA, 3, 146-151. doi:10.1109/IROS.1995.525876

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2021 Preechayasomboon and Rombokas. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.