IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 March 2026, accepted 13 April 2026, date of publication 16 April 2026, date of current version 21 April 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3684279

RESEARCH ARTICLE

SimpleThimble: An Open Source Platform for Wearable Haptics

BERNARDO BROGI[1], (Graduate Student Member, IEEE), LEONARDO FRANCO[1], DOMENICO PRATTICHIZZO[1,2], (Fellow, IEEE), AND GIONATA SALVIETTI[1,2], (Senior Member, IEEE)
[1]Department of Information Engineering and Mathematics, University of Siena, 53100 Siena, Italy
[2]Department of Humanoids and Human Centered Mechatronics, Istituto Italiano di Tecnologia, 16163 Genova, Italy

Corresponding author: Bernardo Brogi (bernardo.brogi@student.unisi.it)

This work was supported in part by European Union's Horizon Europe Program through the Project ''HARIA-Human-Robot Sensorimotor Augmentation'' under Grant 101070292; in part by European Union through the Erasmus+ Program under Grant 2023-1-NO01-KA220-HED-000160462; in part by Beyond the Classroom: Virtual Reality, Augmented Reality and Haptics for Enhanced Surgical Training and Education (ImmersiveSurgicalEdu); and in part by Italian Ministry of University and Research Progetti di Rilevante Interesse Nazionale (PRIN) 2022 ''Augmented Robotic Touch and Hyperspectral Imaging Senses'' (ARTIS) under Grant CUP B53D23002360006.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Local Ethical Committee CAREUS of the University of Siena under Application No. 73/2024.

ABSTRACT This paper introduces the SimpleThimble, an *open-source* project designed to support the development of wearable haptic applications. We detail the device's mechanical and electronic design, provide step-by-step assembly instructions, and explore its role as a possible educational tool. Additionally, we present four software applications that allow users to interact with virtual environments, tailored to different use cases and levels of development expertise. All CAD files, bill of materials, and software are available on the project website. Furthermore, this work shares insights from students who assembled the device as part of short, pilot academic courses at both the Master's and PhD levels. A post-course questionnaire provided initial insights into students' perceptions of the platform as a learning tool, revealing minimal differences between the two student groups despite their background. In addition, a public demonstration with 46 non-expert participants confirmed the device's accessibility and ease of assembly, while a comparative study with a commercial wearable haptic device showed comparable comparable user-perceived feedback quality for the tested interaction task. Together, these results demonstrate the SimpleThimble educational value, usability, and potential as an open platform for haptic learning and research.

INDEX TERMS Wearable haptics, education, open-source.

I. INTRODUCTION

Open source technology has made advanced tools more accessible across various fields. In particular, open-source hardware has driven innovation in areas such as soft robotics [2] and gripper design [3], providing platforms for rapid prototyping and customization.

Beyond their technical impact, open-source devices also support education by enabling hands-on learning experiences, which are crucial in fields like surgery, fine arts, biology, and engineering. To further enhance immersion, Virtual Reality (VR) and Augmented Reality (AR) have been increasingly adopted in education [4], however, without incorporating touch, these platforms lack the realism needed for full engagement [5].

This gap can be addressed through haptic interfaces, which provide force feedback and tactile sensations, enabling users to interact with virtual environments more intuitively. There is an increasing interest in using haptic devices to enhance virtual experiences by providing more sensory information to users [6]. However, most commercial and research haptic devices are too costly and rely heavily on proprietary design, limiting their adoption in educational and research settings [7].

Despite these challenges, open-source haptic solutions have emerged. Examples include Hapkit, developed at Stanford University [7], and OSHap, created by

The associate editor coordinating the review of this manuscript and approving it for publication was Luis Montesinos.

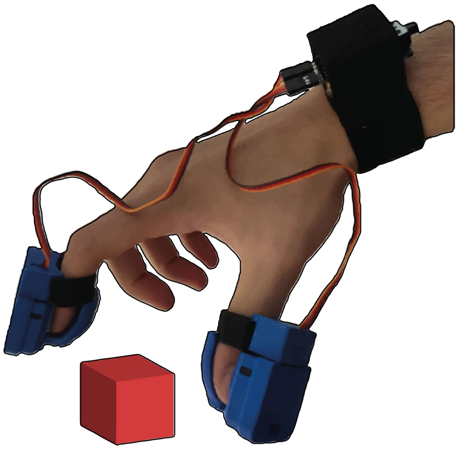


FIGURE 1. The SimpleThimble concept: an open-source, low-cost wearable device that delivers cutaneous force feedback to the fingertip through mechanical transmission actuated by a servomotor, enabling force feedback in virtual reality applications.

Politecnico di Milano [8]. Both devices are desktop haptic interfaces with a limited workspace and a single interaction point, making it impossible to simulate touching an object with all five fingers.

In contrast, wearable haptic devices offer a potentially unlimited workspace and can engage the entire hand, enabling more immersive and natural multicontact interactions [9]. This level of interaction can be achieved through haptic gloves [10]. In general, open-source haptic gloves are characterized by low-cost components, 3D-printable structures, modular electronics, and editable firmware, enabling users to reproduce, repair, and customize devices according to specific application needs. Existing wearable solutions are commonly categorized into thimbles and exoskeletons [11]. Open-source exoskeleton gloves have recently become more available, with representative examples reported in [12], [13], and [14]. Indeed, practical design and fabrication guidelines are now emerging for this class of devices [14]. In contrast, open-source thimble-based platforms remain limited. To the best of our knowledge, Snaptics is one of the few available open-source toolkits for rapidly prototyping wearable multisensory haptic devices [6]. However, Snaptics primarily targets experienced haptic developers rather than educational contexts, and readily reproducible assembly resources are still limited. A qualitative comparison between these representative platforms and SimpleThimble is reported in Table 1.

Building on the need for accessible wearable haptics, we present the SimpleThimble, an open-source, reproducible wearable device that provides cutaneous force feedback at the fingertip via mechanical transmission actuated by a servomotor. Unlike existing toolkits like Snaptics, the device prioritizes accessibility for educators and researchers, focusing on easy assembly and compatibility with virtual environment platforms, allowing adaptation without specialized expertise. The device is depicted in Fig. 1.

Wearable cutaneous haptic devices have advanced the way users perceive and interact with virtual environments. A common approach for generating force feedback in these fingertip devices involves applying normal forces through a single rigid contact plate, which is usually connected to one degree of freedom (DoF). This setup primarily simulates normal or tangential forces through linear motion [15]. Research highlights the importance of fingertip force feedback in replicating realistic tasks. For instance, a device designed by Leonardis et al. [16] showed that applying cutaneous feedback at the fingertip significantly enhances users' control over interaction forces, improving realism and accuracy in task execution. The SimpleThimble builds on this principle by focusing on transient contact rendering, the critical shift from no contact to contact, which is essential for simulating object properties like stiffness [17] and enabling fast-paced interactions such as virtual typing [18].

The SimpleThimble project aims to explore the potential of wearable open-source haptics, focusing on two key aspects. First, it serves as a resource for a broader community of researchers, self-learners, and students interested in wearable haptics as a foundation for developing immersive and realistic VR/AR applications. Second, it provides educators with a tool for designing courses that integrate hands-on experiences through wearable haptic technology. In the rest of the paper, we present the device's mechanical design, electronics, firmware and software, followed by a detailed description of its use in a hands-on course for Master's and PhD students in information engineering. The course served as an illustrative pilot activity aimed at evaluating the SimpleThimble's applicability in educational scenarios, rather than providing a comprehensive assessment of learning outcomes. Moreover, we report the results of a public demonstration with non-expert participants that confirmed the device's accessibility and ease of assembly, while a comparative study with a commercial wearable haptic device showed comparable perceived realism. The paper is structured as follows: Section II describes the SimpleThimble's mechanical design, electronics, hardware assembly and software including four different application scenarios involving different tools for hand and finger tracking, while Section III outlines the educational use of the device in a haptics course and Section IV presents additional evaluations including a public demonstration and a comparative study with a commercial device. Section V discusses the outcomes and evaluation of the experiments with future insights.

II. THE SIMPLETHIMBLE

In this section, we detail the mechanical design, electronics, hardware assembly, and the software developed for the SimpleThimble.

For clarity, we define the "thimble" as the component worn on the hand and the "dongle" as the unit responsible for sending commands to the thimble. This distinction highlights the modularity of the device, as a single dongle can control multiple thimbles simultaneously. In this work, we refer to the *single-thimble* device, designed for poke interactions, where the thimble is worn on the index finger, and the *double-thimble* device, intended for pinch and grasp interactions,

TABLE 1. Qualitative comparison of representative open-source wearable haptic devices and SimpleThimble.

Platform	Form factor	Open-source resources	Assembly guidance	Primary target	Replication effort
DogLove [12]	Exoskeleton glove	Reported open hardware/software resources	Limited in-paper build detail	Research prototyping	Medium/High
Sim et al. [13]	Exoskeleton glove	Reported open implementation resources	Limited in-paper build detail	Research prototyping	Medium/High
McEvoy et al. [14]	Exoskeleton glove	Open resources with design workflow guidance	Structured guidelines reported	Maker and research communities	Medium
Snaptics [6]	Thimble (toolkit)	Open-source prototyping toolkit	Limited ready-to-replicate pathway	Haptic device designers	Medium/High
SimpleThimble (ours)	**Thimble (cutaneous)**	**CAD, electronics, firmware, and software applications**	**Step-by-step assembly and course-oriented material**	**Education and research**	**Low/Medium**

where thimbles are worn on both the thumb and index finger. In both cases, power supply and electronics for control and wireless communication are placed in a bracelet that can be easily worn at the wrist, see Figure 1.

A. MECHANICAL DESIGN

When designing 3D printed parts for haptic devices, it is important to consider the following aspects: the device's ergonomics, the user's comfort, and the device's functionality. The SimpleThimble is designed to be comfortable and easy to wear, with a minimalistic design that intentionally avoids complex geometries to ensure compatibility with entry-level commercial 3D printers. The device consists of three 3D printed functional components: the main body, the rack and pinion mechanism, and the cover. The main body houses the servomotor and the rack and pinion mechanism, while the cover serves as protection from external agents. The rack and pinion mechanism is responsible for converting the rotational motion of the servomotor into linear motion, which is then transmitted to the user's finger through the thimble. The main body and the cover are designed to be easily assembled and disassembled, allowing for easy maintenance and repair. The thimble is designed to be adjustable, allowing it to fit different finger sizes. To attach and fit the user's finger to the device, we used a standard velcro strap, which can be adjusted. The device is designed to be compact (45 mm × 35 mm × 40 mm), lightweight (23 g), and portable, allowing the user to wear it for prolonged periods without discomfort. As discussed in [3], 3D printed part geometries are kept as basic as possible. In this way, the preferred print direction could be easily identified: the main body and the cover can be printed with the fingernail counterpart facing down, while the rack and pinion mechanism with the flat side facing up. This orientation ensures that the parts are printed with the minimum amount of support material, reducing post-processing time and material waste. The parts were printed using a standard FDM (Fused Deposition Modeling) 3D printer (Prusa MK4, Prusa Research a.s., Czech Republic) with a layer height of 0.2 mm and a nozzle diameter of 0.4 mm. We used PLA (PolyLactic Acid) filament with 15% infill density, which provides a good balance between strength and weight and selected for its widespread availability and ease of printing, while adjustable straps accommodate diverse finger sizes without requiring custom modifications.

B. ELECTRONICS AND FIRMWARE

The electronics of this project, as well as all the other components, are designed to be simple and easy to assemble. The SimpleThimble is composed of two main parts: the wearable device and the dongle. The former refers to the thimble that is worn on one or more fingertips and the electronics for the control and the wireless connection that is worn at the wrist together with the battery that provides power supply. The latter is the part used to send commands to the thimble and is normally plugged on a PC. The device is composed of an ESP8266 microcontroller (Wemos D1 mini) with a battery shield that has the same pin footprint, a switch to power on and off the device, and a LiPo battery. The dongle is simply an ESP32 microcontroller development board connected via an USB cable. The ESP8266 microcontroller is used to control the servomotor and receive commands from the dongle, while the ESP32 microcontroller is used to send commands to the thimble. The ESP8266 microcontroller is connected to the servomotor of the thimble through a PWM pin. The ESP8266 microcontroller is powered by a LiPo battery, which is connected to the battery shield. The battery shield is used to charge the LiPo battery and provide power to the ESP8266 microcontroller. The dongle is powered by the USB connection. Fig. 2 illustrates the functional scheme of the SimpleThimble.

The firmware for both the ESP8266 and ESP32 microcontrollers is written in `C++` using the Arduino Integrated Development Environment 2.0. The firmware for the ESP8266 microcontroller is responsible for controlling the servomotor and receiving commands from the dongle, while the firmware for the ESP32 microcontroller is responsible for reading serial data from the USB port and sending commands to the thimble. Codes are designed to be straightforward and easily

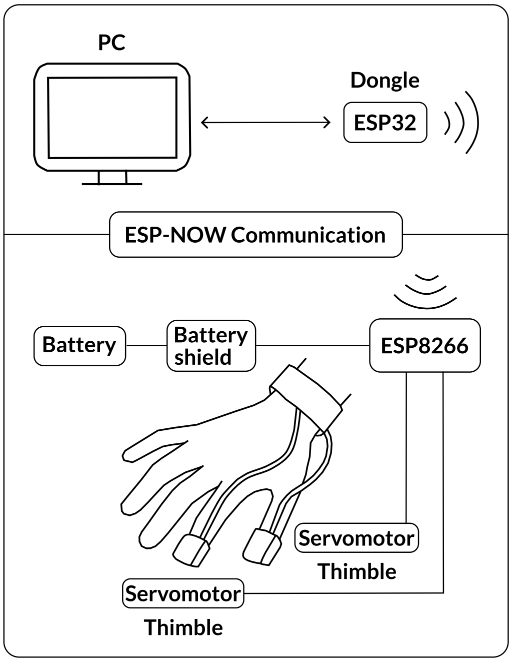


FIGURE 2. Functional block diagram of the SimpleThimble device. The top panel illustrates the dongle connected to a PC, which sends input wirelessly via the ESP-NOW protocol. The bottom panel shows the wearable device worn on the wrist, featuring an ESP8266 microcontroller that receives commands from the dongle and controls the attached servomotors. A battery, connected through a battery shield, powers the system for wearable use.

comprehensible, making them accessible to programmers of all skill levels. The firmware for the ESP8266 microcontroller is designed to be modular, allowing for easy customization and extension. Algorithm 1 outlines the flow of operations that take place in the SimpleThimble firmware.

In this work we considered two thimbles at most, but it is possible to control more thimbles using the same electronics, allowing for a more rich interaction. However, the power supply may need to be upgraded when more than three thimbles are contemporary activated. The firmware for the ESP32 dongle is also modular, allowing for easy customization and extension. The two board communicate with each other relying on the connectionless protocol ESP-NOW which is capable of transmitting data with low latency and high reliability. The ESP-NOW protocol is used to send commands from the dongle to the thimble, allowing for real-time control of the servomotor.

C. HARDWARE ASSEMBLY

For the assembly process of SimpleThimble, we designed the mechatronic components to remain distinct from the electronics, ensuring concise instructions, while allowing users the flexibility to position the electronics and battery as they prefer.

The steps for assembling the SimpleThimble are detailed below:

1) Modify the servomotor horn by trimming its wings to fit the pinion. Secure the horn to the pinion using adhesive or glue.

Algorithm 1 Firmware Logic for Servo Control via ESP-NOW

```
Input: Force data structure incoming stream
Output: Servo motors react to force input data
Initialize servo objects → servo_index, servo_thumb;
Define master MAC address;
Configure servo limits and offsets;
Function OnDataReceive(mac, data, length):
    Copy incoming data → Local force data structure;
    Map force values → Servo angles;
    Update PWM signals → Move servos to desired positions;
Function Setup():
    Initialize serial communication;
    Attach servos → Hardware pins;
    Execute startup sequence → Test servo motion;
    Set device → Wi-Fi Station mode;
    if ESP-NOW initialization fails then
        Display error message → Terminate process;
    Set device role; Add master device → ESP-NOW peer list;
    Register callback → OnDataReceive;
Function Loop():
        ▷ Main loop does nothing → Logic is handled in callbacks
```

2) Insert the rack into the main body, ensuring smooth movement.
3) Slightly tilt the pre-assembled servomotor and pinion while placing them into the main body, aligning the rack and pinion teeth.
4) Attach the top cover to the main body.
5) Secure the back cover onto the servomotor, ensuring the servomotor cable passes through the designated hole.

After having completed these steps, the user should verify that all connections are secure and ensure the device is comfortable to wear before proceeding to the firmware programming and testing phase. Fig. 3 illustrates the assembly process of the SimpleThimble.

Other steps are required to complete the assembly process, such as the addition of neoprene tape for comfort and the addition of the velcro strip to attach the device to the user's finger. Also, the microcontroller should be soldered to its battery shield via a power switch and the male pin header, as well as servomotor cables. The battery just plugs in the charging module which handles its charge phase, signalling the user when it is fully charged with a green led. The charge process is automatically triggered when the user plugs the USB cable to the charging module.

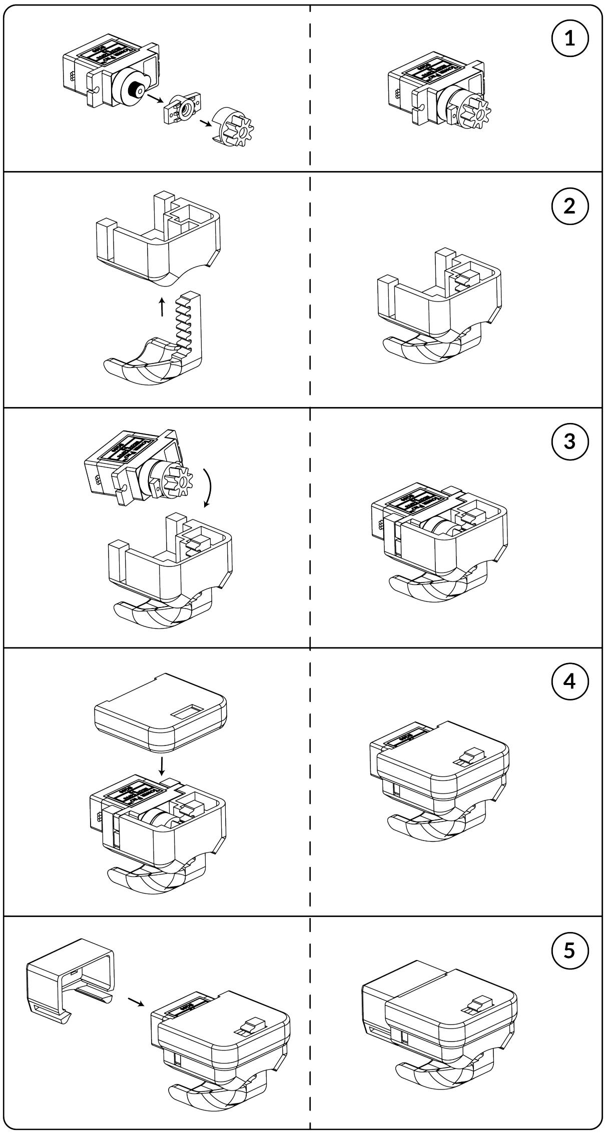


FIGURE 3. **The assembly process of the SimpleThimble is presented in five sequential steps. The illustrations on the left depict the assembly progression, while those on the right showcase the final outcome for each step.**

D. DEVICE CHARACTERISTICS

The SimpleThimble can provide skin indentation at the fingertip. It is relevant to characterise the possible displacement of the platform in contact with the fingertip and the relative exerted force. For a rack and pinion system printed in 3D using PLA filament, the mechanical efficiency η can vary depending on factors such as print quality, surface finish, and lubrication. Based on empirical data and literature [19], a reasonable estimate is:

- Standard print (no post-processing, typical surface roughness): $\eta \approx 0.4 - -0.5$.
- Optimized print (high resolution, post-processing, and lubrication): $\eta \approx 0.6 - -0.7$.

A conservative estimate for most applications with PLA is

$$\eta \approx 0.5\,.$$

The actual force transmitted to the rack, considering friction losses and imperfections introduced by 3D printing, is given by

$$F_{\text{real}} = \frac{2 \cdot T \cdot \eta}{d}\,,$$

where:

- F_{real} is the actual force applied to the rack (in newton, N),
- T is the motor torque (in newton-meters, N·m),
- d is the pinion diameter (in meters, m),
- η is the mechanical efficiency of the system (a value between 0 and 1).

Substituting the values from the stall torque of the servomotor datasheet and considering $\eta = 0.5$, $T = 0.17658\,\text{N}\cdot\text{m}$, $d = 0.01$ m,

the force becomes

$$F_{\text{real}} = \frac{2 \cdot 0.17658 \cdot 0.5}{0.01} = 17.65\,\text{N}$$

The range of displacement S is given by

$$S = \theta \cdot r,$$

where:

- $\theta = 180^\circ = \pi$ rad (angle of rotation),
- $r = \frac{d}{2}$ is the radius of the pinion.

Substituting the values

$$S = \pi \cdot \frac{d}{2}$$

$$S = \pi \cdot \frac{0.01}{2} \approx 0.0157\,\text{m} = 1.57\,\text{cm}.$$

The range of displacement of approximately 1.57 cm allows the device to accommodate for different finger sizes. The servomotor used in the SimpleThimble has a theoretical positional accuracy of 0.1 degrees, that is obviously non appreciable given the great amount of backlash and mechanical inaccuracies of 3D printing. Nonetheless the 1800 possible position allow for fine control of the thimble. The device operates at a speed of 0.12 seconds per 60 degrees, ensuring a responsive and realistic haptic experience. The battery life of the SimpleThimble is approximately 1 hour of continuous use, depending on the intensity and frequency of the force feedback. We tested a 350 mAh battery, however, the device can be used with a larger battery to extend the operating time.

E. SOFTWARE USE CASE

To evaluate the SimpleThimble and provide a range of possible examples of use, four applications were developed using the Unity game engine. These applications, along with a user guide, are publicly available for download on the SimpleThimble website. Their primary goal is to provide intuitive virtual environments for testing the device, each tailored to specific use cases [1]. Figure 4 illustrates the different applications and their setups. The first two applications are designed for educational purposes. They utilize the single-thimble device for poke interactions as shown in Figure 4a. These applications do not consider specific hand-tracking devices, relying solely on mouse and keyboard inputs. This solution avoid potential issues related to variations in tracking systems or software versions and ensure a broad system compatibility. The third and fourth applications are thought for a more advanced use integrating two different hand-tracking systems with the double-thimble device worn on the index finger and thumb (Figure 4b). The third application supports hand tracking via the Leap Motion Controller, while the fourth is compatible with the Meta Oculus Quest head-mounted display. These advanced applications explore the full potential of haptic devices in virtual environments, particularly for grasping digital objects. The Leap Motion Controller and Meta Oculus Quest were selected for their widespread availability and ease of integration.

In all scenarios, the force feedback is computed simulating interaction with rigid objects and it is proportional to the penetration depth of the virtual finger into the object. The formulation of the force feedback is designed to be intuitive and accessible for programmers, allowing for easy customization and extension, as well as providing a starting point for more complex force feedback algorithms. The force feedback is computed using the following equation:

$$F = k \cdot x. \tag{1}$$

Here, F represents the generated force feedback, k is a constant representing the stiffness of the grasped object, and x is the displacement of the virtual finger inside the object being touched or grasped. The displacement is zero when the virtual finger is outside the object and increases as it penetrates, reaching a maximum value equal to half the object's height. The calculated force feedback is transmitted to the device via serial communication through a dongle, which forwards the data to the thimble. The rest of the section details the four virtual reality applications developed for the SimpleThimble.

1) COURSE APPLICATION

The first application provides a testing environment where users interact with a digital cube using a virtual hand controlled by mouse and keyboard. Horizontal movement is controlled via mouse inputs, while vertical movement is managed using keyboard keys. As noted earlier, although this method is less realistic than a hand-tracking system, the application is designed to be accessible to users of all experience levels, including those unfamiliar with haptic devices. When the virtual index finger collides with the cube, the SimpleThimble generates force feedback proportional to the penetration depth of the virtual finger into the cube. This scenario serves as a basic validation tool and is particularly suited for teaching the fundamental concepts of haptics in virtual environments. Furthermore, employing a virtual hand instead of a traditional cursor enhances students' understanding of critical concepts, such as the hierarchical structure of virtual objects and the significance of accurately reproducing feedback within specific locations.

2) DESKTOP APPLICATION

In the Desktop application, users interact with a virtual sphere on a surface populated with obstacles of varying shapes and heights. The sphere, modeled as a rigid body, moves horizontally and is displaced vertically only when pushed against obstacles. To ensure realistic feedback and prevent visual artefacts, a dual-sphere approach is implemented: the *graphic sphere* provides the visual representation, while the *interaction sphere*, invisible to the user, calculates penetration depth based on the intersection volume with obstacles. The generated force feedback is inversely proportional to the obstacle height at the point of intersection. When the sphere collides with an obstacle, the SimpleThimble initially applies the maximum force for that obstacle, which then gradually decreases. This application introduces a more advanced feedback mechanism, enabling users to experience realistic interactions with virtual objects, even when using only a single thimble.

3) LEAP MOTION APPLICATION

The third application involve the Leap Motion Controller SDK, allowing users to interact with and manipulate a digital cube using natural hand gestures. The hand-tracking system enhances immersion by enabling direct control of the virtual hand's movements. To further improve realism, the system incorporates the device featuring two thimbles worn on the thumb and index finger. Despite the presence of this device, the Leap Motion Controller accurately tracks hand movements without compromising the overall user experience. Force feedback is computed based on the penetration depth of the virtual thumb and index finger into the cube, as defined in Eq. 1, and is delivered simultaneously to the corresponding fingers. As previously mentioned, this application is intended for users with a fundamental understanding of haptic devices and hand-tracking systems, offering an engaging and interactive experience. However, users are not fully immersed in the virtual environment; indeed, they observe the scene from an external screen.

4) OCULUS QUEST APPLICATION

In the fourth application, developed for the Oculus Quest (Meta, USA), users interact with a virtual cube using the hand-tracking capabilities of the headset. This fully immersive experience allows users to grasp the digital object through natural hand movements. The application is specifically designed for the two-thimble configuration, enabling users to receive force feedback when performing a pinch grasp. To enhance the realism of force feedback, the system employs the method proposed in [20], where an invisible virtual hand is superimposed on the user's actual hand. The feedback is determined by the penetration depth of this invisible hand into the virtual cube, ensuring a more natural and responsive interaction. This approach maintains the illusion of direct physical contact while accurately conveying haptic sensations. For portability, the application

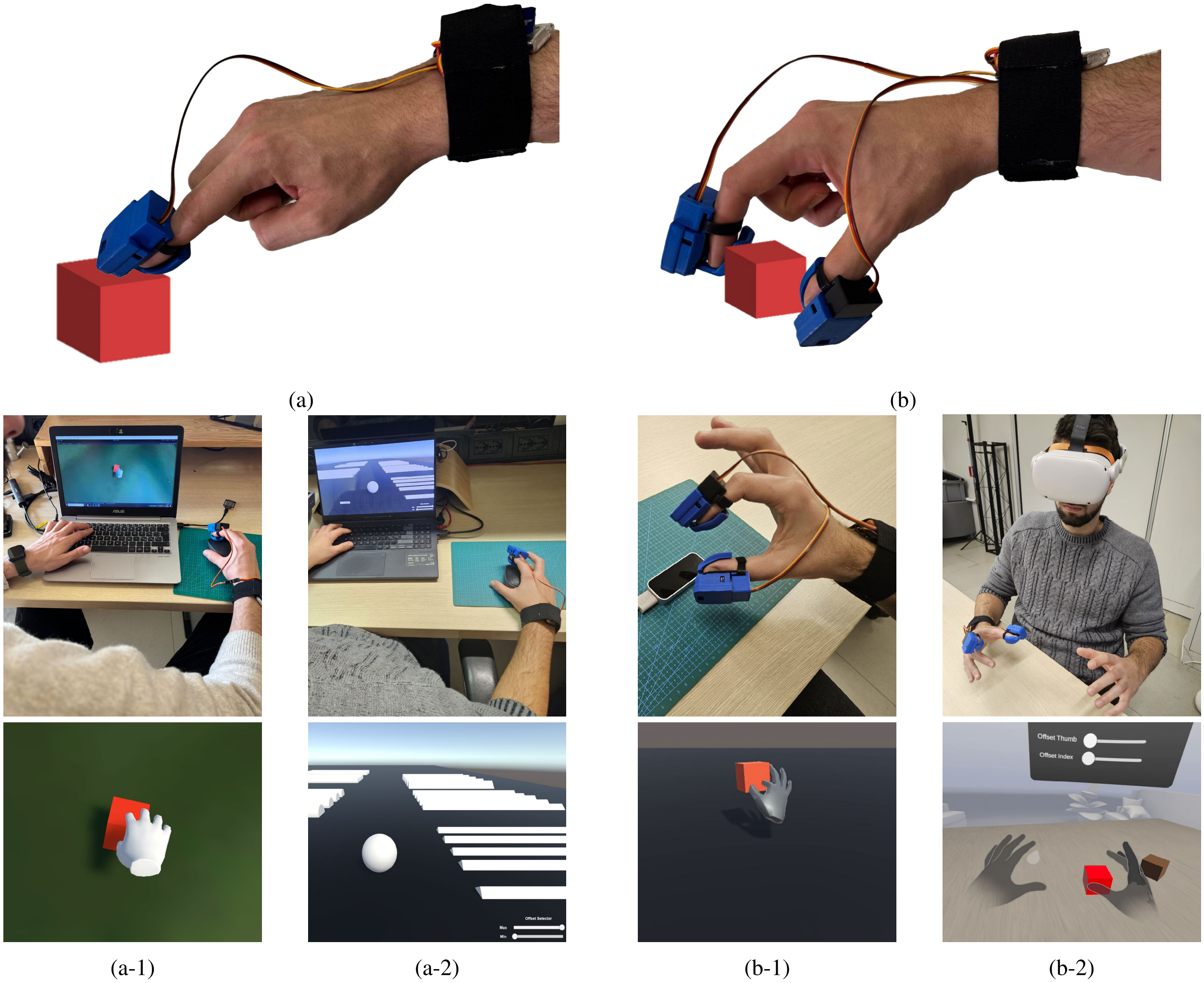


FIGURE 4. **The image consists of three rows. The top row displays the devices used for each application, from left to right: (a) the single-thimble device and (b) the double-thimble device. The second and third rows illustrate the practical use and virtual environments corresponding to each application. From left to right: (a-1) Course Application: A virtual hand is controlled via a laptop mouse and keyboard to interact with a virtual cube, providing force feedback to the index finger through the single-thimble device. (a-2) Desktop Application: A virtual sphere is navigated using a laptop mouse to overcome obstacles, with force feedback perceived in the index finger through the single-thimble device. (b-1) Leap Motion Application: A virtual cube can be grasped by a digital hand that mirrors real hand movements using Leap Motion tracking. Force feedback is delivered simultaneously to the thumb and index finger, and the scene is displayed on a screen. (b-2) Oculus Quest Application: The user can grasp a virtual cube utilizing Oculus hand tracking. Force feedback is applied to both the thumb and index fingers, with the scene viewed directly through the headset for an immersive experience.**

can run as a standalone system on the Meta Oculus Quest. In this setup, Force feedback data is transmitted from the headset to the SimpleThimble via User Datagram Protocol (UDP) communication. To facilitate wireless connectivity, the ESP8266 microcontroller is configured as an access point, allowing the Oculus Quest to connect directly to the SimpleThimble. The force feedback is computed on the headset and sent to the SimpleThimble through UDP packets, eliminating the need for complex Bluetooth configurations. Moreover, a plug-in supporting multiple SimpleThimbles is provided, opening opportunities for researchers to explore more advanced manipulation tasks, such as bimanual interactions. The plug-in is compatible with Unity and can be easily integrated into existing projects, thanks to firmware updates that enable straightforward communication among connected thimbles. Specifically, the communication protocol operates by connecting additional SimpleThimble devices as peers to the main dongle, which manages the transmission of force feedback data to each thimble.

III. EDUCATIONAL USE OF SIMPLETHIMBLE

The SimpleThimble can be used to design introductory courses to wearable haptics. In the following, we report an example of courses attended by master and PhD students in information engineering. The activity was organized as a pilot hands-on experience delivered in two group sessions, with one session attended by 10 PhD students and another by 12 Master's students. The aim of this pilot was to demonstrate the platform usability and accessibility in a real educational setting, rather than to provide a comprehensive evaluation of educational impact.

For Master's students, the activity was embedded in the course *Human-Centered Robotics*, while for PhD students it was embedded in the course *Design of a Wearable Device*. The two courses were delivered in September-October 2024. The Master's course was conducted at the University of Siena, whereas the PhD course was conducted at the Technical University of Munich. The Master's group had a mean age of 23 ± 2 years, while the PhD group had a mean age of 25 ± 2 years.

The course was designed to be accessible to students with a basic understanding of programming in languages such as C, C++, or C#. All participants had a foundational background in programming and prior knowledge of haptics. Our primary objective was to equip students with practical skills in designing and programming wearable haptic devices. Moreover, at the end of the course, students were asked to complete a questionnaire to evaluate the effectiveness of the course and the use of wearable haptic device for teaching.

A. COURSE STRUCTURE AND OVERVIEW

The course spanned approximately three hours, starting with a 30-minute introduction and explanation of the hands-on activity, followed by a 2.5-hour practical session. During each session, students were supervised and guided by the course instructors, who provided assistance and answered questions. The course was divided into three main parts: hardware assembly, firmware programming and testing, and virtual environment integration. Students began by assembling the SimpleThimble, with guidance provided on its components and functionality. All materials, including Arduino scripts and a Unity package, were made accessible through the section *Course* on the public website.

B. HANDS-ON ACTIVITY

Participants were divided into groups: Master's students formed four groups of three, while PhD students were organized into two groups of three and one group of four. Each group received the necessary components to assemble the device, including:

- 3D-printed parts,
- An ESP8266 microcontroller with a battery shield (thimble),
- An ESP32 microcontroller (dongle),
- A servomotor,
- A LiPo battery,
- Accessories such as tapes and strips.

All the equipment is shown in Figure 5.

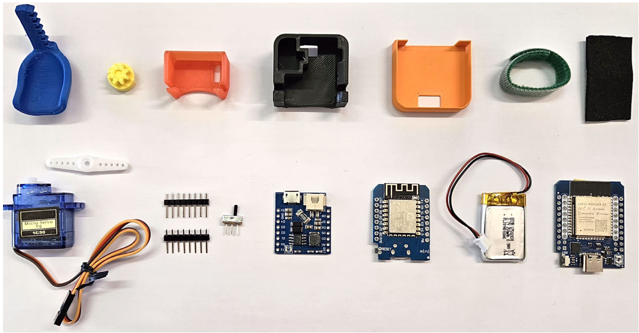


FIGURE 5. The equipment provided to each group for the hands-on activity. These components were used exclusively to demonstrate the function of each individual part, as the electronics were pre-assembled for the practical tasks. Listed from left to right and top to bottom: rack with fingertip envelope, pinion, back cover, main box, box cover, Velcro strip, neoprene tape, servomotor, connection pins, switch, battery shield, ESP8266, LiPo battery, and ESP32.

To maximize the efficiency of the session, components were pre-soldered and 3D-printed beforehand, making the course accessible to students without prior experience in soldering.

Moreover, from a software point of view, students were asked to use their own laptops to upload the firmware and run the application so that we could evaluate the accessibility of the device. In particular, at least one laptop per group was required for the activity. The software prerequisites were given in advance, allowing each student to prepare an appropriate system:

- **Supported operating systems**
  - Windows 10 (64-bit) or later;
  - macOS 10.15 (Catalina) or later;
  - Ubuntu 20.04 LTS (64-bit) or later.
- **Required software**
  - Arduino IDE 2.0 or newer;
  - Unity 2021.3.0f1 (LTS) or newer.

To support a smooth start, installation guides were provided for each platform, and instructors assisted with setup issues at the beginning of the session. During the activity, the tools proved compatible across all operating systems, with students successfully using Windows and macOS machines.

Some highlights of the hands-on activity of PhD students attending the course are shown in Figure 6.

C. FIRMWARE PROGRAMMING AND TESTING

Students were asked to upload the firmware to the devices using the Arduino IDE. The firmware installation process included:

1) Uploading a test firmware to generate rotary motion in the servomotor to verify the device's functionality.
2) Uploading firmware to extract the MAC addresses of the microcontrollers, necessary for establishing wireless communication.
3) Uploading the communication firmware to enable the dongle to wirelessly send input commands to the thimble's servomotor.

Using the Arduino IDE's serial monitor, students tested the communication by sending keyboard ASCII characters to the dongle mapped to motor inputs, observing the corresponding servo motions in the thimble. This process allowed students to verify the correct connection of the devices and establish a baseline for further development.

FIGURE 6. Students testing the device during the hands-on activity. They were divided into groups for the practical session, where they assembled the device, programmed it by uploading firmware, and tested its functionality in a virtual environment.

D. VIRTUAL ENVIRONMENT INTEGRATION

The practical session concluded with a Unity-based exercise in which students imported a Unity package into their project provided from the website dedicated page. This package included scripts, materials, and models for creating a virtual haptic application. The virtual environment featured a cube that could be touched by a virtual hand controlled using the mouse and keyboard of their own laptop. Students implemented scripts to link the collision feedback between the virtual fingertip and the cube to the physical device, enabling haptic responses. During the hands-on session, students worked specifically with the *Course Application* (Figure (a-1)), while the other applications were provided as additional examples for more advanced use.

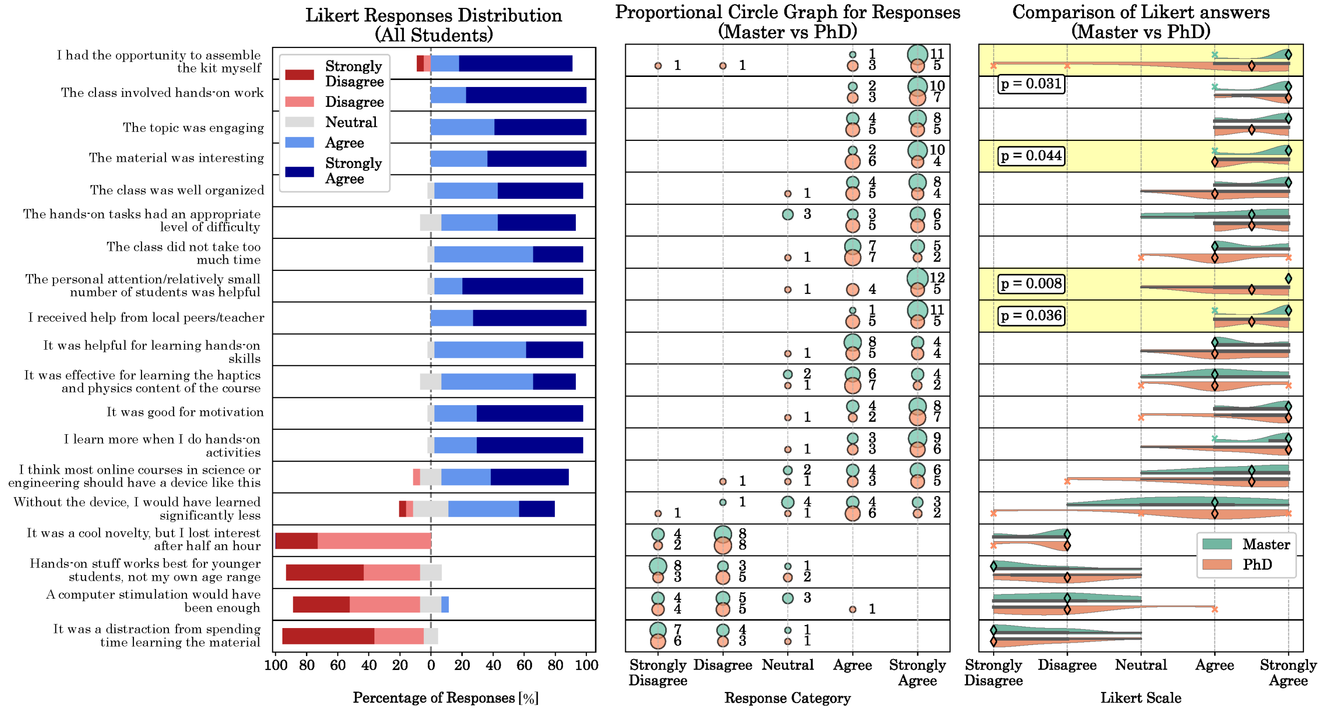
During the exercise, students gained hands-on experience with the Unity game engine and the C# programming language. They learned to detect collisions between virtual objects, compute force feedback values and, establish serial port communication to transmit data to the dongle of the paired device. Additionally, they practised integrating basic keyboard and mouse inputs to interact with the virtual environment.

E. SURVEY RESULTS

All students successfully completed the session, assembling the device, uploading the firmware, and integrating the SimpleThimble within a virtual environment. The time allowed for each task was sufficient, with students completing the assembly in approximately 30 minutes, firmware programming in 45 minutes, and virtual environment integration in 1 hour and 15 minutes. These time frames included troubleshooting and debugging as well, ensuring that all students could progress at a comfortable pace.

A post-course questionnaire was administered anonymously to evaluate the session's effectiveness, featuring Likert-scale questions adapted from a pre-validated survey of [21]. The questionnaire was modified from the original one to reflect the in-person nature of the course, which did not include online modules. The answers corresponded to a 5-point scale, ranging from 1 (strongly disagree) to 5 (strongly agree). The feedback highlighted the success of the hands-on approach and identified factors contributing to the learning experience. As this was an exploratory pilot activity, questionnaire-based perceptions were used as preliminary indicators of feasibility and acceptance, and not as standalone evidence of competency acquisition. Approval of all ethical and experimental procedures and protocols was granted by the local Ethical Committee CAREUS of the University of Siena (No. 73/2024), and all participants provided informed consent. The survey investigated the potential factors contributing to success in the course, including the opportunity to hands on experience, the organization of the course, and the interest in the topic. Moreover, in the second part of the questionnaire, the students were asked to evaluate the effectiveness of the course in terms of the overall satisfaction and the usefulness of the practical session. Outcomes were collected for both groups of students, and results were first aggregated (Figure 7a) to provide an overall view of student feedback. In general, students reported a positive experience with the course. High levels of agreement were observed for questions related to motivation, the usefulness of hands-on tasks, and the perceived value of the practical session. Most students agreed that the physical device supported their learning, that the class was well organized, and that the tasks had an appropriate level of difficulty. Notably, both groups indicated that the device was neither unnecessary nor distracting, and there was no reported decline in attention or interest over time. Although both Master's and PhD students shared a background in information engineering, we chose to analyze their responses separately to explore whether differences in academic maturity and learning expectations influenced their perceptions of the course. To this end, a non-parametric Mann-Whitney U test was performed [22]. The resulting p-values (p) quantify whether observed differences between the two groups are statistically significant ($p < 0.05$).

From the outcomes (Figure 7b), a significant differences emerged in perceptions of opportunities to assemble the kit ($p = 0.031$) and the helpfulness of personalized attention ($p = 0.008$). PhD students appeared to value personalized support more than master's students, possibly due to higher expectations for advanced, individualized learning experiences. Conversely, master's students may have prioritized active participation, reflecting their developmental stage in learning. Moreover, a significant difference was also found in responses to receiving help from peers or teachers ($p = 0.036$). PhD students appeared to rely more on expert guidance to complement their self-directed learning, whereas master's students seemed to benefit from peer interactions. Another significant difference was observed in the perception of the material's interest level ($p = 0.044$). PhD students, with potentially greater background knowledge, might find certain material less novel, which could affect their level of interest. Most other aspects, including perceptions of task difficulty, organization, motivation, and learning effectiveness, showed no significant differences between the groups. This uniformity indicates that the course was broadly effective across both groups. However, some trends merit further attention. Both groups valued hands-on activities similarly,



(a) (b)

FIGURE 7. Analysis of survey results collected from students after their haptics course experience. The figure displays three types of graphs that share the same y-axis, listing the survey questions, where (a) shows the overall distribution of responses, and (b) illustrates differences between groups. The graph on the left (a) shows the distribution of responses for all students combined, expressed as percentages. To distinguish between agreement and disagreement, the right side of the bars indicates agreement, while the left side indicates disagreement. The central graph in (b) is a circle plot, representing the distribution of responses for each question by group, Master's (green) and PhD (orange). Larger circles indicate a higher number of responses. The graph on the right in (b) is a violin plot, illustrating the distribution of Likert scale responses for the two groups. The violin shape represents response density, with median values marked by diamonds and outliers indicated by 'X' symbols. Survey questions showing statistically significant differences between groups ($p < 0.05$) are highlighted with a yellow background, emphasizing key variations between Master's and PhD students.

yet neither reported a significantly enhanced learning experience compared to other methods ($p > 0.4$ across related questions). This aligns with prior research suggesting that hands-on approaches are most effective when explicitly aligned with intended learning outcomes and conceptual instruction [23], [24]. In our case, the practical activities were intentionally mapped to the core class contents. Indeed, assembly was linked to device architecture and mechanical transmission concepts, firmware tasks were linked to communication and control concepts (ESP-NOW, PWM, serial mapping), and the Unity exercise was linked to interaction modeling through Eq. 1. This indicates a substantial theory-practice alignment at the task-design level, although future iterations should include more explicit guided reflection to further strengthen conceptual transfer.

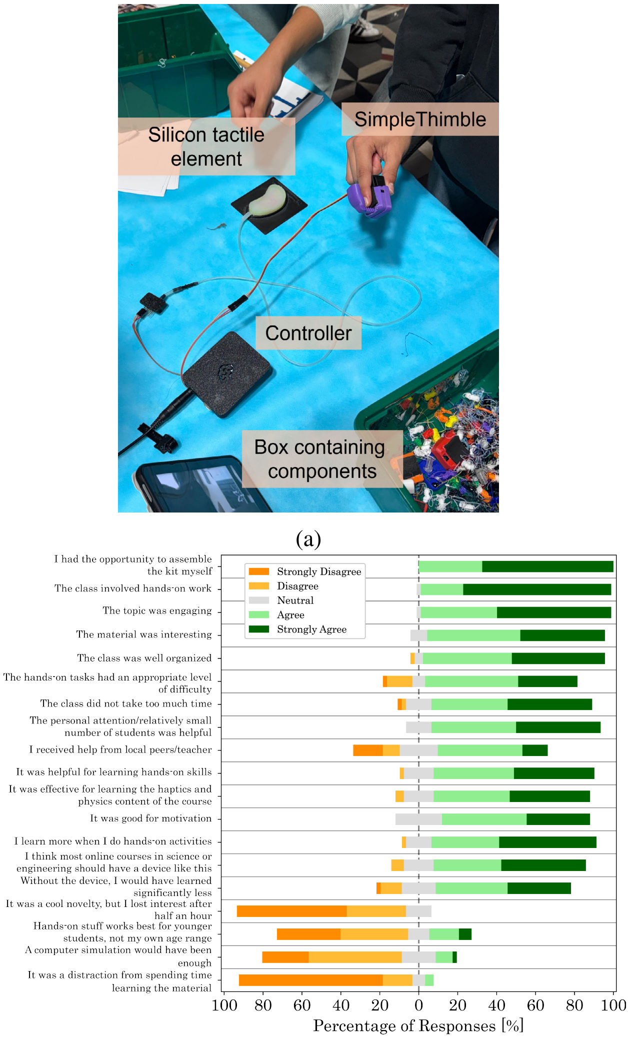
IV. ADDITIONAL EVALUATIONS

To further validate the accessibility and realism of the SimpleThimble, two complementary experimental activities were conducted beyond the pilot academic course in order to assess the interaction with a non-expert audience and the performance compared to a validated commercial haptic device.

A. PUBLIC DEMONSTRATION

The activity tailored for non-expert participants was carried out during a public science event. Attendees were invited to assemble their own device, without any programming, and use it as a haptic interface to render the pressure of a silicone tactile element connected to an MPX5050 pressure sensor. Participants pressed the tactile plate and perceived a corresponding force on the thimble, which was modulated according to the measured pressure. During the demonstration, participants had to select the the correct components from a box containing both relevant and non-relevant items, following assembly instructions. Whenever the device was correctly assembled, they could test it by connecting to the controller and pressing the tactile element. After the experience, participants were asked to fill out a short survey similar to the one presented for the course but adapted for the event. 46 people participated in the demonstration (26 females, 20 males, mean age 16 ± 3), all without prior experience in haptics. A picture of the demonstration together with the results of the survey is shown in Fig. 8.

The survey results show that participants greatly appreciated the hands-on nature of the demonstration, particularly the opportunity to assemble the device themselves and



(a)

(b)

FIGURE 8. **(a) Participants assembling and testing the SimpleThimble during a public science event. The device was used to render the pressure applied to a silicon tactile element, allowing users to experience haptic feedback in an interactive manner. (b) Results of the survey conducted among 46 participants after the demonstration.**

immediately test it in a real application scenario. Moreover, most respondents rated the materials and content as engaging and motivational, confirming that the system's simplicity and ease of use were clear and inspiring to a broad audience even if non-expert.

Beyond its educational purpose, this demonstration was designed as an illustrative use-case inspired by telemedicine. Recent studies highlight the growing interest in telepalpation systems that aim to support telemedicine by allowing clinicians to "feel" tissues remotely, improving diagnostics and remote patient monitoring. For instance, novel wearable devices incorporate sensing mechanisms that measure and transmit tactile palpation data from a patient to a doctor [25], enabling the remote perception of skin pressure through haptic interfaces. Our demonstration reproduced this concept in a simplified form suitable for a public event, using a silicone tactile element and a pressure sensor to simulate the sensation of finger pressure. This setup was intended only to illustrate a possible application domain for wearable haptic devices, rather than to validate the SimpleThimble for telemedicine or clinical palpation.

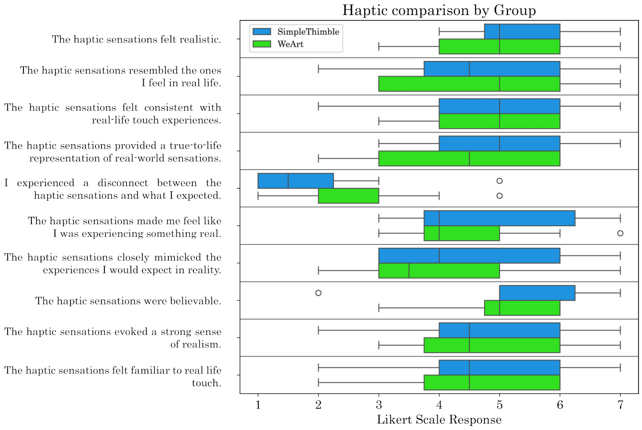


FIGURE 9. **Comparison between the SimpleThimble and the WEART TouchDiver based on user feedback collected through a 7-point Likert scale questionnaire.**

B. COMPARISON WITH A COMMERCIAL DEVICE

An additional user study was conducted to compare the performance of the SimpleThimble with a commercial wearable haptic device, the TouchDiver (WEART, Italy). The study involved 12 participants (6 males, 6 females, all right handed, mean age 27 ± 3). The evaluation was performed using the *desktop application* described in Sec II-E. For a fair comparison, both devices were used with only one thimble worn on the index finger, and the order of device use was counterbalanced across participants. During the experiment, participants explored a virtual environment using both the SimpleThimble and the TouchDiver, each worn on the index finger and operated together with a mouse. Pink noise was played throughout the trials to minimize distractions and maintain focus. After completing the exploration with each device, participants rated a 7-point Likert scale questionnaire, using items adapted from [26]. Fig. 9 shows the results of the questionnaire.

To compare the two devices, a Mann-Whitney U test was performed for each questionnaire item. The results revealed no statistically significant differences between the SimpleThimble and the TouchDiver across all items, suggesting that our proposed device provides a comparable user experience for the intended task. These findings highlight the effectiveness of the SimpleThimble in delivering realistic haptic feedback within its design scope.

V. DISCUSSION

A. OUR FINDINGS

Based on the survey results and overall feedback from the practical course, several key insights were obtained regarding both the hardware requirements and their arrangement within the course. The SimpleThimble was well received by students, who appreciated the opportunity to assemble and program a wearable haptic device. Its simplicity and modularity allowed them to focus on core haptic design concepts without being overwhelmed by complex hardware. Although some hardware and software setup issues emerged during the pilot activities, the modularity of the device, along with the choice of open-source software compatible with multiple operative systems, ensured a flexible and accessible learning experience. Furthermore, opting for serial ports and ESP-NOW communication instead of Bluetooth proved beneficial for the course, simplifying the setup and reducing potential connection issues. The questionnaire results also highlighted areas for improvement. A key take home message was the value of assembling the device. Among master's students, 91.6% strongly agreed and 8.4% agreed that they had the opportunity to assemble the kit themselves. However, the experience varied across participants since students worked in groups. PhD students, being more accustomed to working independently, had more opportunities for autonomous work, while master's students collaborated more in groups, leading to a more varied experience. These differences suggest that course design should better accommodate the varied expectations and backgrounds of students. However, fully customizing the course for PhD students remains challenging due to their diverse backgrounds and research topics. Another important observation was the perceived impact of the device on learning. While the majority of students agreed that it enhanced their understanding, a small percentage (8.3% of master's and 10% of PhD students) disagreed. This may be due to prior familiarity with the topic or the group-based format, which occasionally limited individual hands-on experience. Differences also emerged in students' interest in the material and their perception of the supervisor's support. Master's students, less accustomed to working with physical devices (83.3% strongly agreed, 16.6% agreed), found hands-on activities particularly valuable. In contrast, PhD students, who probably have more opportunities to engage with hardware (40% strongly agreed, 60% agreed), were less dependent on these experiences. Similarly, while master's students highly valued the supervisor's support (100% strongly agreed), PhD students, being more independent, found it less essential. Overall, the course was well received in terms of topic relevance, organization, difficulty level, and time allocation. Both groups found the time slots for each task appropriate. However, some groups faced software issues that made the allocated time for firmware programming too short. In contrast, the 30-minute time slot for assembling was longer than necessary for certain groups. This discrepancy suggests that future courses should be more flexible in terms of time allocation, allowing for individual differences in learning speed and technical proficiency. Both groups agreed that it was not a distraction from other activities and contributed positively to their motivation. The hybrid approach of combining theoretical learning with practical activities proved effective in maintaining student engagement, despite differences in expertise and background. These insights highlight the SimpleThimble promise not only as a reliable and effective educational tool but also, as confirmed by the additional demonstration and haptic comparison, as a versatile platform for broader applications. In particular, the public demonstration confirmed the device's accessibility for non-expert users, who were able to assemble and use it effectively, showcasing its potential beyond academic settings. Moreover, the comparison with a commercial device further validated the SimpleThimble's capability to deliver a comparable user experience, underscoring its potential as a cost-effective alternative for wearable haptic applications.

B. LIMITATIONS

The present study has several limitations that should be considered when interpreting the results. First, the educational evaluation was conducted as a pilot experience with a limited number of participants, all of whom were graduate students in information engineering. This small and homogeneous sample reduces the generalizability of the findings across academic levels, disciplines, and educational contexts.

Second, the course evaluation relied primarily on post-activity self-reports. While these measures are informative for assessing usability, engagement, and acceptance, they do not provide a rigorous assessment of learning outcomes on their own. No objective external measures, such as written or oral examinations, grading of practical assignments, or controlled comparisons with alternative teaching conditions, were included.

Third, the group-based format led to a non-uniform hands-on experience. Although collaborative work supported the activity, it also meant that opportunities for individual assembly and direct interaction with each stage of the process were not identical across participants. Relatedly, learners' sense of ownership and the impact of customization were observed qualitatively but were not directly measured through the survey.

Fourth, the evaluation framework focused mainly on usability, engagement, and comparative user perception, and did not include dedicated haptics-oriented psychophysical experiments. As a result, claims about perceptual realism cannot be supported in a rigorous way, since no controlled measurements such as just-noticeable differences [27] or discrimination thresholds were collected. For the same reason, the telemedicine-related scenario should be interpreted as an illustrative demonstration of possible application domains rather than as validation for clinical palpation or medical use. From a technical perspective, the chosen low-cost actuation strategy, based on a servomotor-driven rack-and-pinion transmission, inherently introduces mechanical limitations such as backlash, friction, and possible cogging effects. These factors can reduce rendering fidelity and transparency, especially in applications requiring precise force reproduction. In addition, the simplified and customizable assembly process may introduce variability across devices, further complicating the acquisition of consistent psychophysical measurements.

Finally, the pilot also exposed practical constraints of the current platform, including occasional cable disconnections, soldering-related issues, and variability in software and driver configuration when participants used their own laptops. While the comparison with a commercial device showed promising results for the intended task, the current SimpleThimble design is limited to force feedback and does not yet support additional haptic modalities such as temperature or texture rendering.

C. FUTURE WORKS AND PERSPECTIVES

The previous limitations highlight several ways to improve the reliability and robustness of the device. From an hardware perspective, in the next version of the SimpleThimble, a protective cover for the wearable microcontroller and a structured cable management system will be introduced. Recent advancements in microcontroller technology have already enabled a significant reduction in the size of the electronics needed for the device. For example, an ESP-based microcontroller which integrates battery management and has the size of a coin such as the *Seeed Studio XIAO* ESP32C6 [28] could be used. This will allow for a more compact and comfortable design, enhancing the user experience. Also the size of the motors could be decreased by choosing a smaller servomotor, which requires an update of the mechanical design. From an electronics and sensing perspective, the firmware will be upgraded to incorporate new features, including force sensing capabilities (integrating force sensors like the Force Sensitive Resistance). These enhancements will enable more precise control and richer haptic feedback, further improving the versatility of SimpleThimble for educational and research applications. Since SimpleThimble showed promising potential as a learning tool, several improvements can be made to enhance the course structure and better accommodate the diverse needs of students. Future research should aim to include participants from a wider range of backgrounds to more rigorously assess the applicability and impact of the SimpleThimble in diverse domains. While our initial course served as a proof of concept, it would be particularly valuable to extend the experience to undergraduate students and external educators. To ensure a more consistent and engaging learning process, students should have more opportunities for individual assembly. Assigning specific roles, such as assembly, programming, and testing, within each group, along with estimating the time required for each task, can help maintain active participation and optimize time management.

This will also lead to the integration of the SimpleThimble within full-semester courses, where learning outcomes can be evaluated through objective assessment methods aligned with curricular competencies. These methods will include written and oral examinations, project and practical task evaluations, structured faculty observations during manipulation tasks, and comparisons with control groups or alternative instructional conditions.

Future survey instruments could also be expanded to explicitly measure perceived ownership, motivation [29], and the impact of customization, offering a more rigorous evaluation of these pedagogical outcomes.

Building on the course's success, it can also be expanded in order to cover more advanced topics such as CAD design, 3D printing, and virtual environment development. These additions could enable students to customize elements of the SimpleThimble, such as adjusting mechanical dimensions for finger fit, modifying firmware parameters, or extending Unity-based interactions.

VI. CONCLUSION

In this work, we presented the SimpleThimble, an open-source, low-cost wearable haptic device designed to support educational activities and integration into virtual reality environments. The device features a simple mechanical structure and a modular design, enabling easy assembly and customization. It is equipped with a servomotor that delivers force feedback to the fingertip, enhancing immersion in virtual interactions. To ensure accessibility, the SimpleThimble is accompanied by a comprehensive documentation, including firmware, 3D-printable models, and virtual reality integration packages, all hosted on a dedicated website. The platform is compatible with various operating systems and cross-platform software, making it flexible and broadly usable. The device was validated in an pilot educational setting through a hands-on course involving both Master's and PhD students in information engineering, as well as through public demonstrations and a comparative evaluation with a commercial wearable haptic device to further assess the performance. By capitalizing on its reproducibility through modular hardware and cross-platform software, the SimpleThimble positions itself as a foundational tool for research and education in robotics, wearable haptics, and immersive technologies.

REFERENCES

[1] SimpleThimble Project. (2026). *SimpleThimble Documentation and Resources*. Accessed: Mar. 12, 2026. [Online]. Available: https://simplethimble.readthedocs.io

[2] D. P. Holland, C. Abah, M. Velasco-Enriquez, M. Herman, G. J. Bennett, E. A. Vela, and C. J. Walsh, "The soft robotics toolkit: Strategies for overcoming obstacles to the wide dissemination of soft-robotic hardware," *IEEE Robot. Autom. Mag.*, vol. 24, no. 1, pp. 57–64, Mar. 2017, doi: 10.1109/MRA.2016.2639067.

[3] R. Ma and A. Dollar, "Yale OpenHand project: Optimizing open-source hand designs for ease of fabrication and adoption," *IEEE Robot. Autom. Mag.*, vol. 24, no. 1, pp. 32–40, Mar. 2017, doi: 10.1109/MRA.2016.2639034.

[4] F. Sanfilippo, T. Blazauskas, G. Salvietti, I. Ramos, S. Vert, J. Radianti, T. Majchrzak, and D. Oliveira, "A perspective review on integrating VR/AR with haptics into STEM education for multi-sensory learning," *Robotics*, vol. 11, no. 2, p. 41, Mar. 2022. [Online]. Available: https://www.mdpi.com/2218-6581/11/2/41

[5] J. Fröhner, G. Salvietti, P. Beckerle, and D. Prattichizzo, "Can wearable haptic devices foster the embodiment of virtual limbs?" *IEEE Trans. Haptics*, vol. 12, no. 3, pp. 339–349, Jul. 2019, doi: 10.1109/TOH.2018.2889497.

[6] Z. A. Zook, O. O. Ozor-Ilo, G. T. Zook, and M. K. O'Malley, "Snaptics: Low-cost open-source hardware for wearable multi-sensory haptics," in *Proc. IEEE World Haptics Conf. (WHC)*, Jul. 2021, pp. 925–930, doi: 10.1109/WHC49131.2021.9517172.

[7] M. O. Martinez, J. Campion, T. Gholami, M. K. Rittikaidachar, A. C. Barron, and A. M. Okamura, "Open source, modular, customizable, 3-D printed kinesthetic haptic devices," in *Proc. IEEE World Haptics Conf. (WHC)*, Jun. 2017, pp. 142–147, doi: 10.1109/WHC.2017.7989891.

[8] A. Lavatelli, F. Ferrise, and M. Bordegoni, "Design of an open-source low cost 2DOF haptic device," in *Proc. IEEE/ASME 10th Int. Conf. Mech. Embedded Syst. Appl. (MESA)*, Sep. 2014, pp. 1–6, doi: 10.1109/MESA.2014.6935574.

[9] C. Pacchierotti, S. Sinclair, M. Solazzi, A. Frisoli, V. Hayward, and D. Prattichizzo, "Wearable haptic systems for the fingertip and the hand: Taxonomy, review, and perspectives," *IEEE Trans. Haptics*, vol. 10, no. 4, pp. 580–600, Oct. 2017, doi: 10.1109/TOH.2017.2689006.

[10] O. Ozioko and R. Dahiya, "Smart tactile gloves for haptic interaction, communication, and rehabilitation," *Adv. Intell. Syst.*, vol. 4, no. 2, Feb. 2022, Art. no. 2100091. [Online]. Available: https://advanced.onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202100091

[11] J. Perret and E. V. Poorten, "Touching virtual reality: A review of haptic gloves," in *Proc. 16th Int. Conf. New Actuat. (ACTUATOR)*, 2018, pp. 1–5.

[12] H. Zhang, S. Hu, Z. Yuan, and H. Xu, "DOGlove: Dexterous manipulation with a low-cost open-source haptic force feedback glove," 2025, *arXiv:2502.07730*.

[13] D. Sim, Y. Baek, M. Cho, S. Park, A. S. M. S. Sagar, and H. S. Kim, "Low-latency haptic open glove for immersive virtual reality interaction," *Sensors*, vol. 21, no. 11, p. 3682, May 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/11/3682

[14] A. McEvoy, N. Murray, and C. Keighrey, "Building a low-cost immersive haptic feedback glove for virtual reality," in *Proc. 16th Int. Conf. Quality Multimedia Exper. (QoMEX)*, Jun. 2024, pp. 58–61.

[15] A. Frisoli and D. Leonardis, "Wearable haptics for virtual reality and beyond," *Nature Rev. Electr. Eng.*, vol. 1, no. 10, pp. 666–679, Sep. 2024, doi: 10.1038/s44287-024-00089-8.

[16] D. Leonardis, M. Solazzi, I. Bortone, and A. Frisoli, "A 3-RSR haptic wearable device for rendering fingertip contact forces," *IEEE Trans. Haptics*, vol. 10, no. 3, pp. 305–316, Jul. 2017, doi: 10.1109/TOH.2016.2640291.

[17] K. J. Kuchenbecker, J. Fiene, and G. Niemeyer, "Improving contact realism through event-based haptic feedback," *IEEE Trans. Vis. Comput. Graphics*, vol. 12, no. 2, pp. 219–230, Mar. 2006, doi: 10.1109/TVCG.2006.32.

[18] N. T. Tanacar, M. H. Mughrabi, A. U. Batmaz, D. Leonardis, and M. Sarac, "The impact of haptic feedback during sudden, rapid virtual interactions," in *Proc. IEEE World Haptics Conf. (WHC)*, Jul. 2023, pp. 64–70, doi: 10.1109/WHC56415.2023.10224429.

[19] O. Doğan and M. S. Kamer, "Wear characteristics of 3D-printed spur gears: Material type and design parameters effects," *Iranian Polym. J.*, vol. 34, no. 9, pp. 1–15, Sep. 2025, doi: 10.1007/s13726-024-01442-2.

[20] B. Brogi, G. Cortigiani, A. Villani, N. D'Aurizio, D. Prattichizzo, and T. L. Baldi, "The avatarm: Interacting in the physical metaverse via robotics, diminished reality, and haptics," *IEEE Access*, vol. 12, pp. 90750–90767, 2024, doi: 10.1109/ACCESS.2024.3420717.

[21] T. Morimoto, M. Orta Martinez, R. Davis, P. Blikstein, and A. M. Okamura, "Teaching with hapkit: Enabling online haptics courses with hands-on laboratories," *IEEE Robot. Autom. Mag.*, vol. 28, no. 3, pp. 79–91, Sep. 2021, doi: 10.1109/MRA.2020.3024395.

[22] J. D. Winter and D. Dodou, "Five-point Likert items: T test versus mann-whitney-wilcoxon," *Practical assessment*, vol. 15, no. 11, pp. 1–12, 2010, doi: 10.7275/bj1p-ts64.

[23] J. Biggs, "Enhancing teaching through constructive alignment," *Higher Educ.*, vol. 32, no. 3, pp. 347–364, Oct. 1996, doi: 10.1007/bf00138871.

[24] M. J. Prince and R. M. Felder, "Inductive teaching and learning methods: Definitions, comparisons, and research bases," *J. Eng. Educ.*, vol. 95, no. 2, pp. 123–138, Apr. 2006, doi: 10.1002/j.2168-9830.2006.tb00884.x.

[25] M. Pompilio, N. D'Aurizio, T. L. Baldi, L. Franco, G. Gabriele, and D. Prattichizzo, "A novel wearable sensing device enabling remote palpation," in *Proc. IEEE Haptics Symp. (HAPTICS)*, Apr. 2024, pp. 149–156, doi: 10.1109/HAPTICS59260.2024.10520853.

[26] T. Shi and O. Schneider, "Development and initial validation of the haptic experience inventory (HXI)," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, Apr. 2025, pp. 1–21, doi: 10.1145/3706598.3713834.

[27] S. Allin, Y. Matsuoka, and R. Klatzky, "Measuring just noticeable differences for haptic force feedback: Implications for rehabilitation," in *Proc. 10th Symp. Haptic Interfaces Virtual Environ. Teleoperator Syst. (HAPTICS)*, 2002, pp. 299–302, doi: 10.1109/haptic.2002.998972.

[28] Seeed Studio. (2024). *Seeed Studio XIAO ESP32C6*. Accessed: Feb. 3, 2025. [Online]. Available: https://www.seeedstudio.com/Seeed-Studio-XIAO-ESP32C6-p-5884.html

[29] K. Ciampa, "Learning in a mobile age: An investigation of student motivation," *J. Comput. Assist. Learn.*, vol. 30, no. 1, pp. 82–96, Feb. 2014, doi: 10.1111/jcal.12036.

BERNARDO BROGI (Graduate Student Member, IEEE) received the B.Sc. degree in computer engineering and the M.Sc. degree (cum laude) in artificial intelligence and automation engineering from the University of Siena, Siena, Italy, in 2020 and 2022, respectively. He is currently pursuing the joint Ph.D. degree in robotics and intelligent machines (national interest) with the University of Genoa and the University of Siena. His research interests include robotics and extended reality, focusing on human–robot collaboration for healthcare and wellness of persons.

LEONARDO FRANCO received the M.Sc. degree in biomedical engineering from Roma Tre University, Italy, in 2018, and the Ph.D. degree from the University of Siena, Italy, in 2023. From 2023 to 2025, he was a Researcher with Siena Robotics and Systems Laboratory (SIRSLab), University of Siena, where he also the Head of the Mechatronic Development for projects in haptics and human–robot interfaces.

DOMENICO PRATTICHIZZO (Fellow, IEEE) received the M.S. degree in electronics engineering and the Ph.D. degree in robotics and automation from the University of Pisa, Pisa, Italy, in 1991 and 1995, respectively. In 1994, he was a Visiting Scientist with the MIT AI Laboratory. He is currently a Professor of robotics with the University of Siena, Siena, Italy. Since 2009, he has been a Scientific Consultant with Istituto Italiano di Tecnologia, Genova, Italy. He has authored more than 200 articles in his research fields. His main research interests include haptics, grasping, visual servoing, and mobile robotics.

GIONATA SALVIETTI (Senior Member, IEEE) received the master's degree in computer engineering and the Ph.D. degree in information engineering from the University of Siena, Italy, in 2009 and 2012, respectively. From 2012 to 2015, he was a Postdoctoral Researcher with Italian Institute of Technology, Genoa, Italy. He is currently an Associate Professor with the Department of Information Engineering and Mathematics, University of Siena, and an Affiliated Researcher with Italian Institute of Technology. His research interests include collaborative robotics, assistive devices, and haptics.