eTactileKit: A Toolkit for Design Exploration and Rapid Prototyping of Electro-Tactile Interfaces

Praneeth Bimsara Perera*
School of Computer Science
The University of Sydney
Sydney, NSW, Australia
ppat0685@uni.sydney.edu.au

Ravindu Madhushan Pushpakumara*
School of Computer Science
The University of Sydney
Sydney, New South Wales, Australia
Department of Electronic and Telecommunication Engineering
University of Moratuwa
Colombo, Western, Sri Lanka
ravindu.pushpakumara@sydney.edu.au

Hiroyuki Kajimoto
The University of Electro-Communication
Chofu, Japan
kajimoto@uec.ac.jp

Arata Jingu
Saarland University
Saarland Informatics Campus
Saarbrücken, Germany
jingu@cs.uni-saarland.de

Jürgen Steimle
Saarland University, Saarland Informatics Campus
Saarbrücken, Germany
steimle@cs.uni-saarland.de

Anusha Withana
School of Computer Science
The University of Sydney
Sydney, NSW, Australia
Sydney Nano Institute
The University of Sydney
Sydney, NSW, Australia
anusha.withana@sydney.edu.au

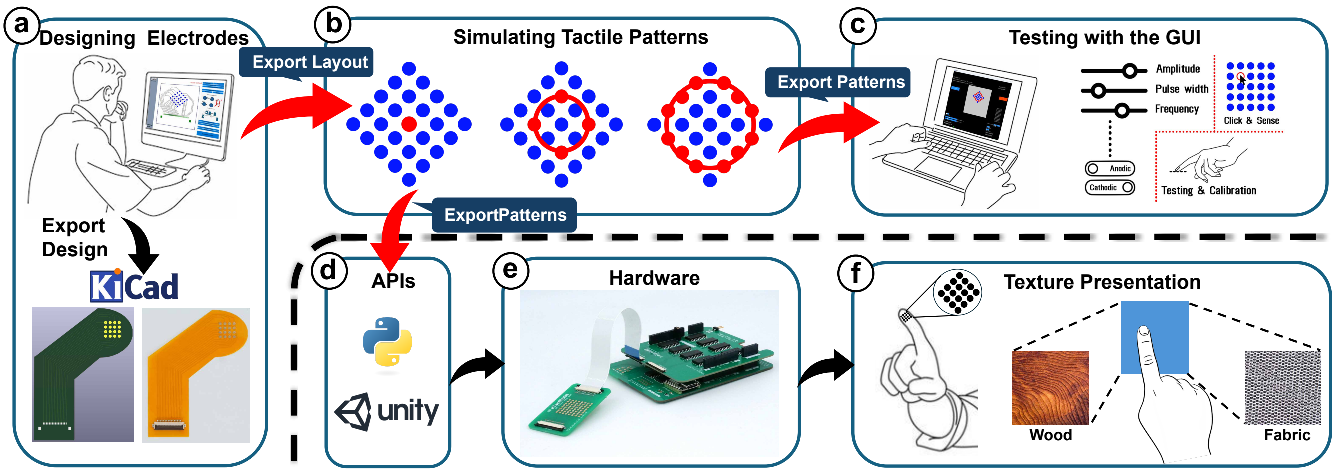

Figure 1: Designing a finger-worn electrode interface for different texture presentations using eTactileKit. The designer creates electrode layouts for fabrication using the (a) electrode interface design tool, (b) program and visually simulates the tactile patterns on the designed layouts, and tests and explores patterns on (e) hardware using the (c) interaction tool GUI. Finally, created patterns are synchronously sent to hardware using the (d) APIs in an application to (f) present tactile feedback resembling various textures when interacting with a smooth surface.

Abstract

Electro-tactile interfaces are becoming increasingly popular due to their unique advantages, such as delivering fast and localised tactile response, thin and flexible form factors, and the potential to create

*Both authors contributed equally to this research.

UIST '25, Busan, Republic of Korea

ACM ISBN 979-8-4007-2037-6/25/09
https://doi.org/10.1145/3746059.3747796

novel tactile experiences. However, insights from a formative study with typical designers highlighted the lack of resources, limited access to information and complexity of software and hardware tools. This establishes a high barrier to entry and limits the ability to rapidly prototype and experiment with electro-tactile interfaces. To address these challenges, we propose eTactileKit, a scalable and accessible toolkit providing end-to-end support for designing and prototyping electro-tactile interfaces. eTactileKit comprises a hardware platform and a software framework for designing, simulating and exploring electro-tactile stimuli. We evaluated the impact and usability of eTactileKit through a three-week long take-home study, which demonstrated increased accessibility, ease of use, and the toolkit's positive impact on design workflow. Additionally, we implemented a set of use cases to demonstrate the toolkit's practicality and effectiveness across various applications.

CCS Concepts

• **Human-centered computing → User interface toolkits**; **Haptic devices**.

Keywords

Electro-tactile, Haptics, Toolkit, DIY, Rapid prototyping

ACM Reference Format:

Praneeth Bimsara Perera, Ravindu Madhushan Pushpakumara, Hiroyuki Kajimoto, Arata Jingu, Jürgen Steimle, and Anusha Withana. 2025. eTactileKit: A Toolkit for Design Exploration and Rapid Prototyping of Electro-Tactile Interfaces. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25), September 28–October 01, 2025, Busan, Republic of Korea.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3746059.3747796

1 Introduction

Electro-tactile interfaces stimulate the sensory nerve fibers in the skin with frequency-modulated electrical currents to create tactile sensations [17]. Compared to vibrotactile mechanisms, electro-tactile stimulation is increasingly becoming popular among researchers due to several advantages, including faster response time and higher bandwidth [17], better localization [10, 44], and ability to support deformable form factors due to the lack of need for rigid and bulky mechanical actuators [4, 8, 47, 51, 54]. Additionally, electro-tactile interfaces can be made using 3D [8, 33] and 2D [12, 54] printing, enabling customisation and rapid prototyping. Furthermore, electro-tactile interfaces have been shown to create novel tactile experiences, such as rendering coldness [39], softness [11, 46], and preserving the user's natural tactile sensation while delivering additional tactile feedback [49–51, 54]. These benefits have made electro-tactile ubiquitous in diverse application domains [12, 16, 18, 19, 31, 32, 45, 47].

Despite these advantages, prototyping and tinkering with electro-tactile interfaces remain comparatively challenging compared to other tactile stimulation methods. Electro-tactile interfaces need electrical circuitry that operates at relatively high voltages with precise current-control mechanisms to assure safety and comfort under changes in skin impedance [13, 17]. They also need accurate frequency and pulse width control along with multiplexing different channels, which further complicates hardware and firmware designs [14, 51, 54]. Furthermore, unlike other tactile interfaces, the stimulation interfaces in electro-tactile are electrode arrays that need to be self-designed to match the application requirements, such as shapes, density, and sizes, while complying with functional needs of operations (e.g., discrimination thresholds) [44, 47]. Collectively, these challenges have made electro-tactile interfaces largely inaccessible to novice designers, limiting the potential of electro-tactile interfaces to create novel tactile experiences.

To this end, we propose eTactileKit, an open-source toolkit for electro-tactile interface design tailored to assist developers with different levels of expertise. It comprises hardware, firmware, and software with design tools for exploration and rapid prototyping, orchestrated to support all stages of electro-tactile interface design. We informed the design of eTactileKit from a formative study with 7 participants who had experience working with electro-tactile-related research projects. The thematic analysis from the formative study resulted in four main themes, each embedding a set of requirements for the toolkit based on participants' comments.

To satisfy these requirements, eTactileKit comprises (1) a reconfigurable and scalable hardware design to control electro-tactile stimulation and handle electrode interfacing, (2) an electrode interface design tool, (3) a simplified scripting and visualisation interface to simulate and program different tactile experiences, (4) an interaction tool with a Graphical User Interface (GUI) to execute patterns and real-time exploration of sensations, (5) a set of application programming interfaces (APIs) to operate the toolkit from popular platforms (Python and Unity 3D), and (6) comprehensive documentation on using each of the aforementioned components and guidelines for getting started with the toolkit.

We qualitatively evaluate eTactileKit's impact and usability from a longitudinal study with 8 participants who designed applications and incorporated eTactileKit into their electro-tactile projects. The results from our study confirm that eTactileKit lowers the barrier to entry, is easy to use, and positively impacts both novice and experienced designer workflows, enabling rapid prototyping and design exploration. Finally, we demonstrate the capabilities of eTactileKit by implementing a set of applications that confirm the toolkit's potential and its usefulness for diverse domains.

In summary, the contributions of this paper are:

- A formative study with 7 participants highlighting challenges faced by designers, suggestions, and a set of requirements for a toolkit for electro-tactile.
- eTactileKit, an open-source toolkit designed to assist at all stages of electro-tactile interface design, comprising hardware, firmware, and software tools for design exploration and rapid prototyping, which is validated through a longitudinal study.
- Set of applications developed using eTactileKit demonstrating the toolkit's potential and usefulness.

2 Related Work

In this section, we discuss the background of electro-tactile interfaces, their advantages and the challenges involved. Moreover, we discuss the importance of toolkits in HCI and highlight prior toolkit work with both hardware and software components and how they have helped the research community.

2.1 Electro-Tactile Interfaces and Applications

Electro-tactile systems operate by delivering controlled electrical stimulation to the skin, emulating various tactile sensations without the need for physical actuators [12, 17, 20, 51]. These systems use electrodes placed on the skin surface to apply small electrical currents [18, 54], which activate the underlying sensory nerves. By fine-tuning parameters such as intensity, frequency, and waveform, a wide range of sensations, including pressure [57], vibration [57], and texture [5, 17] can be generated. Moreover, it has been shown that electro-tactile systems can deliver sensations such as softness [11, 46, 48], tastes [26] and coldness [39]. Apart from the above abilities, the advantages of electro-tactile systems include their ability to produce a wide range of tactile sensations without needing physical actuators, making challenging form-factors possible [51, 54]. Additionally, these systems can be highly responsive and energy-efficient [57].

These advantages enable electro-tactile interfaces to be ideal solutions in many different application areas. These includes, teleoperation in robotics [31], assisting visually impaired people [16, 19], and advancing user experience in virtual reality [12, 32, 45] and augmented reality [18], and enhancing mobile device experience [47].

2.2 Electro-Tactile Systems and Challenges

Despite the advantages of electro-tactile interfaces, the challenges that need to be addressed to enable wider adoption of the technology are multifaceted. Firstly, designers need to have a deep understanding of current control and its corresponding implementation in hardware. Delivering a current through the high impedance of the skin requires a high-voltage generation circuit followed by a current-driver circuit, which enables constant current pulses at higher impedances [14]. Moreover, electro-tactile interfaces require this current to be precisely controlled in terms of pulse height (intensity) and pulse width (duration of elicitation), which needs advanced current controlling mechanisms using high frequency switching circuits [17, 54] and continuous impedance measurement to adjust the current accordingly [13]. Additionally, the frequency of activation of the electrodes with accurate timing is vital, as it affects the perception of the tactile sensation. It also requires individual access to the electrodes, which is achieved by switching (multiplexing) circuits. The aforementioned hardware should be accurately controlled by carefully programmed firmware.

Beyond the control electronics, further challenges lay in designing the layout of the electrode interface to comply with parameters such as the minimum discrimination threshold of different skin regions to electro-tactile stimulation[44, 47]. Furthermore, the calibration process of the electro-tactile interfaces is not easy, as it requires the designer to understand the impedance relationship of the skin [17] and design the electrode-skin interface to deliver the desired tactile sensation within safe limits of current. Apart from pulse parameters, the electro-tactile experience relies upon created electrode activation patterns (tactile patterns) [11, 48], which can become a cumbersome procedure to test and program to achieve the desired tactile pattern.

Even though prior research has informed the community on the design of electro-tactile interfaces [13, 14, 17, 54], the complexity of the hardware and software components, and the need to understand electronics and physical concepts, hinder the wider adoption of the technology. Therefore, an openly accessible toolkit that encompasses and hides the aforementioned complexities is essential to enable wider adoption and innovation with the technology.

2.3 Toolkits in HCI

Greenberg [6] defines a toolkit from a programmer's perspective as a way of encapsulating interface design concepts such as widget sets, interface builders, and development environments. Moreover, Oulasvirta and Hornbæk [29] define toolkit research as "producing understanding about the construction of an interactive artifact for some purpose in human use of computing". Therefore, toolkits [7, 35, 37, 41, 53, 55, 58] can be seen as enablers for designers to mitigate the complexity of underlying concepts and enable a more streamlined approach to achieve objectives, enabling wider adoption of the technology, saving iteration time [22, 43].

In the electro-tactile research space, Tactlets [8] provides a toolkit to rapidly prototype electrodes on 3D surfaces with integrated touch sensing, yet does not provide hardware and software assistance on other components of the electro-tactile interface design. Similarly, TactTongue [26] provides a toolkit for tactile feedback on the tongue, yet it does not assist hardware compatible with widely adopted electro-tactile interfaces that require higher currents [51, 54]. Thus, given the diversity of the application space, an extensible and modular toolkit that can provide end-to-end support for electro-tactile interfaces would enable wider adoption of the technology and motivate innovation in electro-tactile research.

3 Formative Study

In order to bring forth eTactileKit with accessibility to both novice and experienced designers, we conducted a formative study with 7 participants who have engaged with electro-tactile-related projects. The study included semi-structured interviews to understand the challenges faced by designers and developers in getting started with electro-tactile projects, the resources available to them, and the areas they thought could be improved in their technical workflow.

3.1 Method

3.1.1 Participants. The participants included 6 males and 1 female, aged 20 to 39 years ($M = 25.86$, $SD = 3.87$) and were not compensated for participation. Participants had background experience in Computer Science (n = 5) and engineering (n = 2). The inclusion criteria for the participants was to have experience with at least one electro-tactile-related project. Background information revealed that participants are working in diverse application areas for electro-tactile. For instance, participants mentioned changing object sensation to alter tactile perceptions and pseudo-haptic applications that simulate physical interactions in virtual environments. Other notable applications include enhancing teleoperation in robotics and integrating electro-tactile in 3D printed objects.

3.1.2 Semi Structured Interviews. The interviews were conducted online via the Zoom video-conferencing application, and each took approximately 30 to 45 minutes. They were audio and video recorded with the consent of the participants. We asked a set of open-ended questions that were designed to understand the participants' background information, their experience with electro-tactile-related projects, the challenges they faced in getting started with electro-tactile, and the resources available to them in terms of learning materials, hardware, and software. We further inquired about the challenges or areas they thought could be improved in their technical workflow, and additional technical support that would have helped to streamline the workflow and design better applications. At the end, participants were allowed the opportunity to add further comments or ask questions.

3.2 Thematic Analysis

All the interviews were transcribed and anonymised for further analysis. The transcripts were then manually refined word-by-word to identify any errors. Two authors conducted a thematic analysis to analyse the data, following Braun and Clarke's approach [3]. The thematic analysis followed: (1) independent open coding, (2) refining codes together, (3) identifying themes, and (4) refining themes. The thematic analysis resulted in 4 main themes that we identified as the challenges and requirements for developing eTactileKit. A detailed elaboration of participant experience, comments and supportive evidence from participants is provided in Appendix A.

3.2.1 Theme 1: High Barrier to Entry. The first theme to emerge was the *high barrier to entry* in developing and experimenting with electro-tactile interfaces. This challenge is primarily attributed to the lack of publicly accessible resources, documentation, and other user-friendly knowledge sources. Most participants expressed that they relied on personal networks, such as colleagues or supervising academics, for guidance and resources to get started with electro-tactile technology. Research papers were their main source of information, but they often found them complex and difficult to apply in practical contexts. Moreover, the lack of easily accessible and user-friendly resources was identified as a significant hurdle for exploring the technology, particularly for newcomers or those outside specialised academic backgrounds, such as electronics.

3.2.2 Theme 2: Enabling Design Exploration. The second theme that emerged from the thematic analysis was the significant lack of tools and methods available for iteratively exploring designs in electro-tactile technology. Unlike other tactile feedback methods, which benefit from well-developed, easy-to-use tools, electro-tactile design requires substantial fundamental work. Participants emphasised that designers often spend more time developing the underlying technology, as outlined in theme 1, than on the creative process of designing tactile experiences. The lack of accessible tools for developing hardware, particularly for designing electrode interfaces and control hardware, was highlighted as a significant challenge. Furthermore, participants highlighted the need for software tools and APIs to create tactile patterns, handle communication, and control stimulation parameters (stimulation intensity, frequency, etc.). The absence of GUI tools to quickly explore the design space was also noted. These findings underscore the need for more comprehensive and accessible tools, both hardware and software, to support the development of electro-tactile technology.

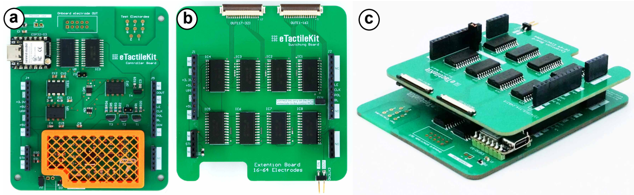

Figure 2: Control hardware for eTactileKit: (a) the main controller module, (b) the switching module, and (c) a perspective view of the switching module stacked on top of the main controller module.

3.2.3 Theme 3: Extensibility and Scalability. The third theme that emerged from the analysis was the lack of methods to extend and scale electro-tactile interfaces. Participants highlighted that there are no straightforward ways to increase the number of electrodes without completely redesigning the hardware, making it difficult to experiment with more complex or larger-scale designs. Similarly, changing the form factor of devices presents significant challenges due to the rigid nature of current hardware designs. Moreover, most software and communication APIs, particularly firmware, are developed with little consideration for scalability. These findings suggest the requirement for extensible and scalable hardware and software solutions to accommodate various design requirements and application complexities.

3.2.4 Theme 4: Safety and Calibration. The fourth theme that emerged from the analysis was the pressing need for support and assurance regarding fundamental necessities in electro-tactile technology. Participants expressed concerns about safety, emphasising the importance of reliable methods to minimise errors, especially given the direct interaction with the human body and high voltages. Furthermore, safety concerns in electro-tactile systems were raised during the programming of stimulation patterns. This can be overcome by system calibration to deliver comfortable yet noticeable tactile stimuli. Participants highlighted the need for user-friendly software-based calibration methods to expedite the calibration process and ensure consistent and safe results.

3.3 Identified Requirements for eTactileKit

Following the thematic analysis, we identified a set of requirements for the development of eTactileKit. The requirements were derived from the generated themes and participant comments during the interviews. The requirements are presented in Table 1.

4 eTactileKit

Based on the identified requirements from our formative study, we developed eTactileKit, a modular and extensible toolkit for designing and implementing electro-tactile interfaces. In this section, we describe eTactileKit and how its components assist designers in the electro-tactile design process by fulfilling the requirements R1-R4. The components described include the modifications made after the final user study. The complete toolkit is open-sourced, and all the resources are available on the eTactileKit GitHub repository[1].

Table 1: Identified requirements for eTactileKit and their descriptions

R1. Design tool for designing electrode layouts	
R1a. Design 2D electrode layouts	Allow quick and easy customisation of electrode PCBs and electrodes in 2D surfaces
R1b. Design 3D electrode layouts	Ability to modify 3D surfaces to electro-tactile interfaces
R2. GUI software for testing and exploration	
R2a. Creating stimulation patterns	Tool to easily create electro-tactile stimulation patterns
R2b. Simulation of patterns	Ability to simulate the programmed patterns prior to executing them on hardware
R2c. Basic stimulation pattern library	Basic set of ready-to-use stimulation patterns
R2d. Streamlined calibration procedure	Calibration tools that expedite the calibration process and ensure consistent and safe results.
R2e. Real-time monitoring electrode states	Allows real-time monitoring of electrode states and sensing.
R2f. Interaction with electrodes	Allows interaction with each electrode individually
R3. Hardware design	
R3a. Safety ensured easy-to-use hardware	Safer-to-handle hardware modules avoiding custom hardware design.
R3b. Inbuilt set of electrodes to test functionality	An inbuilt set of electrodes on board enabling testing full system functionality without any user involvement on electrode design
R3c. Scalable hardware	Hardware that supports easy scaling for the number of electrodes.
R3d. Extensible firmware	Firmware supporting different modes of operation and configurations for electro-tactile stimulation
R3e. Reliable communication with hardware	Ability to easily communicate with hardware and firmware that support scalability as application complexity increases.
R3f. API for other platforms	API to access hardware through Unity and support for languages like Python
R4. Documentation	
R4a. Quick Start Guide	A quick start guide to get started with the toolkit
R4b. Comprehensive Guidance	Documentation on how to use each of the toolkit components.
R4c. Handling guidelines	Set of handling guidelines and tools to ensure safe operation and minimise errors.

4.1 eTactileKit Hardware (R3)

While electro-tactile interfaces have been extensively studied, there remains a scarcity of accessible, open-source hardware platforms for rapid prototyping and experimentation. Existing circuit designs are often proprietary, and their replication requires specialised expertise in electronics and PCB design, creating a significant barrier to entry for non-specialists. To address this, eTactileKit offers a DIY hardware platform for designers to design and test various electro-tactile-based experiences with basic electronic skills. The PCBs are designed to be easily soldered by designers with minimal soldering skills and resources by using a sparse component layout and the use of through-hole components when possible instead of surface mount devices (SMD).

Prior works have used project-specific hardware designs, hindering their ability to be scaled or adapted to different applications. eTactileKit hardware is designed to be modular and easily scalable, with the help of a main controller module (see Figure 2-a) and a switching circuit module (see Figure 2-b) that can be easily stacked on top of each other (see Figure 2-c). This modular design allows designers to easily customise the number of electrodes supported, enabling them to scale the hardware to suit their specific needs and requirements (R3c). Each switching module is responsible for connecting external electrodes and can handle up to 64 electrodes. The number of electrodes can be increased to 128 by stacking up 2 switching modules and so on. The switching module uses 4 FPC connectors, each supporting 16 channels, to connect electrodes as shown in Figure 1-e. Moreover, debugging is important to ensure proper connections and isolate any issues involved with the designs. Thus, to facilitate quick testing of tactile patterns and validation of hardware, the main controller module has 8 built-in electrodes that allow designers to explore tactile patterns without connecting any external electrodes (R3b), enabling it to function similarly to onboard LEDs on the Arduino platform.

Electro-tactile hardware has to support configurations based on application requirements [11] and region of stimulation on the body [44] (also see Appendix B). Therefore, the main controller module is designed to support a wide range of configurations adopted by prior works. The module requires a high voltage to drive currents through the skin's higher impedance. Several studies have adopted power supply voltages in the range of 200V to 300V [14], and we use 270V. This supply enables a voltage-to-current controller to deliver currents regardless of the variations in skin impedance. This can deliver pulse height (current intensity) per channel within a range of 0–3.8mA, which can be incremented in 200×19μA steps. This range

[1]https://github.com/aid-lab-org/eTactileKit

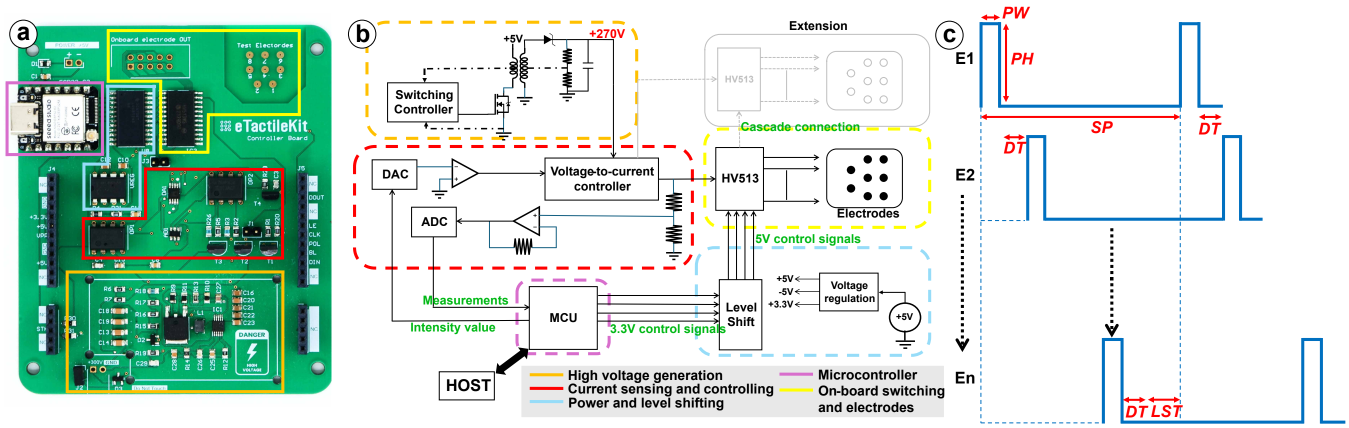

Figure 3: A dissected view of the main controller module: (a) annotated view of the main circuit units of the main controller board. (b) Simplified schematic block diagram of the circuit. (c) Timing diagram of an electrode activation scenario where all n electrodes are activated.

is decided based on the requirements of prior work [26, 40, 49, 54]. However, these are soft limits and can be adjusted in firmware up to 10mA maximum current and 2.44μA minimum current step. This satisfies applications that require higher current intensities [51]. Each electrode is individually activated using temporal multiplexing using a Microchip HV513 multiplexer [25]. The system uses a stimulation frequency of 1–250Hz and a pulse width that can be incremented in 1μs steps.

Figure 3-c shows a timing diagram of **n** electrodes. All **n** electrodes are activated in this case. The intensity of the pulse is represented by the pulse height **(PH)**, and the duration of stimulation is the pulse width (**PW**). The inter-channel activation delay (**DT**) is set to discharge the channels before activating the next channel. The time taken to activate all **n** electrodes is $(\mathbf{PW} + \mathbf{DT}) \times \mathbf{n}$. This time should be less than the total round-trip time available for the multiplexer to activate all the electrodes, **SP**. The stimulation frequency is determined by $\frac{\mathbf{1}}{\mathbf{SP}}$. A loop stabilisation time **LST** is a delay added to maintain the target frequency. Thus, the maximum number of electrodes that can be used in a given stimulation parameter configuration is determined to satisfy the above criteria. The communication with hardware is done with bytes at a baud rate of 921600 bits/s, which takes around 11μs per byte. Therefore, the effect of USB throughput would be minimal in a practical application scenario with a few hundred electrodes.

Furthermore, the controller module is designed to accommodate calibration and comprises extensible firmware (R3d) to allow different configurations of electrodes, precise timing of control operations, and modes of operation (anodic and cathodic). During anodic stimulation, one electrode acts as the anode (current sourcing), and the rest act as cathodes (current sinking). Thus, by default, the stimulation is monophasic and can be configured to biphasic AC stimulation by alternating between Anodic and Cathodic modes, satisfying the requirements shown by [26, 51].

A microcontroller (Seeed Studio XIAO ESP32S3) handles control operations and manages communication with the host device. Moreover, the firmware ensures reliable communication with the hardware without obstructing the stimulation operation. This is achieved by implementing the control and communication tasks on separate threads using FreeRTOS (R3e).

4.1.1 Hardware Implementation. The schematic block diagram of hardware is shown in Figure 3-(a, b). This circuit is derived based on prior work [13, 15, 56]. The main controller module is powered by 5V from the microcontroller. A Switching Voltage Regulator (MAX1044CPA+) is used to convert 5V to −5V, which creates a dual rail power supply for the OP-Amps. It has an 8-bit level shifter (SN74LVC8T245) to convert 3.3V control signals from the microcontroller to 5V control signals required by the switching circuitry.

High voltage is generated by a flyback power supply is implemented using a DC/DC controller (LT3757) to convert the 5V DC input to a high voltage 270V DC output. Our circuit implementation is based on the reference design of the LT3757 datasheet.

The current drive is implemented using a voltage-to-current converter. The required voltage is generated by a 12-bit Digital-to-Analog Converter (AD5452), which is then converted into current by a current multiplier circuit similar to the one described in [14]. The output voltage across the current driving circuit is fed to a voltage divider, and the voltage drop is read by a 12-bit Analog-to-Digital Converter (AD7276) connected to the microcontroller. This reading is used to compute the impedance across the electrodes.

4.1.2 Safety Features. To ensure safety during operation, the hardware implements several safety features. A current limiting diode (Semitec E-103, current limit 10mA) is used to prevent excessive current flow in case of a fault. This component can be replaced with other diodes with different current ratings, depending on the application requirements (see Appendix B). The current driving circuit delivers constant currents set by the DAC output, regardless of skin impedance variations. This prevents any overdraining of current. Furthermore, the DAC will be disabled if the microcontroller stops working, stopping the current flow. Moreover, to ensure safety while handling the hardware, the high voltage carrying traces are

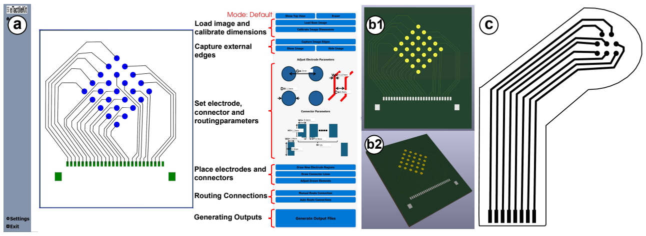

Figure 4: eTactileKit's 2D electrode design tool: (a) annotated tool layout with an example design scenario of a tactile display with 25 electrodes, (b1) 3D top view and (b2) perspective view of the PCB visualised in KiCAD. (c) 1:1 scale PDF export for finger-worn electrode interface.

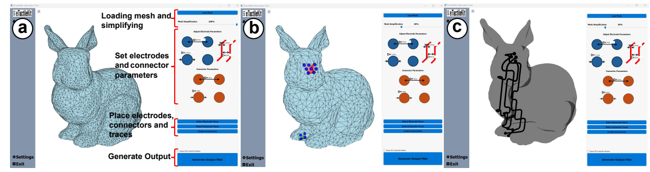

Figure 5: eTactileKit's 3D electrode interface design tool used to design an electrode interface on the Stanford Bunny surface. (a) The mesh is loaded to the tool, (b) electrodes are populated on the nose and connectors between the front legs. (c) The 3D view of the connections routed from electrodes to the corresponding connector.

localised to a specific region (R3a) and enclosed with a 3D printed protective cover (see Figure 2-a) to avoid accidental contact.

4.2 Electrode Interface Design Tool (R1)

Electrodes are the main interface for an electro-tactile system. They can be 2D [51, 54] or 3D [8] surface electrode interfaces. However, designing electrode interfaces is challenging due to several factors. Firstly, electrodes should be manually designed considering the design parameters of electrodes (size, gap between electrodes), decided based on the tactile acuity of the region of the body where the stimulation is intended. This can become a cumbersome process when the number of electrodes increases, especially for 3D surface electrode layouts. Next, implementing connections between electrodes and the hardware should be properly managed for reliable operation and easy control of electrodes when interfacing with hardware. This requires implementing connectors that need to be individually routed to the electrodes. Moreover, accomplishing these requires prior knowledge and skills in PCB design, 2D illustration or 3D modelling software, depending on the intended type of interface designed.

Consequently, eTactileKit provides an electrode interface design tool developed using Python that allows designers to design custom electrodes on both 2D (R1a) and 3D (R1b) surfaces. Furthermore, the tool hides the inbuilt complexity of electrode interface design and provides a simplified interface to design custom electrode interfaces without requiring prior knowledge of specific software. This could even benefit experienced designers, lowering their design iteration time, as highlighted in the formative study.

4.2.1 Electrode Design on 2D Surfaces. Figure 4-a shows the 2D electrode interface design mode of the tool. Firstly, the designer can import an image to the canvas, which has the desired outline for the electrode interface. The image should then be calibrated to the required dimensions, and the tool can capture the edges of the image to define the outline polygon of the electrode interface. Designers can then define the set of parameters, which includes the size and spacing of the electrodes, connector parameters and routing parameters. Next, regions can be selected on the canvas by free-hand drawing polygons, and the tool will automatically populate each region with electrodes using the specified parameters. Connector locations can be defined by drawing line segments at the desired orientation, and the tool generates connectors along each line segment. The tool ensures that all generated electrodes and connectors are within the bounds of the outline polygon. To finalise the design, the tool supports both automatic and manual routing of the connections between the electrodes and the connectors. The automatic routing is done using the A* algorithm to find the shortest path between two points on a grid, avoiding obstacles in between.

Prior work has used 2D electrode interfaces created by multiple manufacturing methods, including inkjet printing [54], laser cutting [11], and PCB fabrication [18, 26, 51]. To accommodate this, the tool supports the design exports of custom electrode interfaces that can be fabricated using any of these methods. For the above fast-prototyping techniques, the tool can export 2D vector layouts as PDF files with a one-to-one scale of the design (see Figure 4-c) and masks for electrode locations to assist during fabrication. For PCB design, the tool exports as a JSON file with all the necessary electrical element properties that can be imported into KiCAD, an open-source PCB design software, using a plugin developed with Python (see Figure 4-(b1, b2)). The designers can refine the design further if required and export the Gerber files to manufacture the PCB. Moreover, it should be highlighted that prior work incorporates PCBs more often due to their durability and physical properties to withstand usage conditions [26, 51]. Thus, we provide the option to use a standard FPC connector [23] footprint for the PCB design mode, the same as the one used in the switching circuit module, for seamless integration with the hardware. In addition to the design files generated from the tool, the electrode interface design tool separately exports a JSON file containing the electrode locations and captured outline coordinates to interface the electrode designs with the subsequent software in the eTactileKit.

4.2.2 Electrode Design on 3D Surfaces. Figure 5 shows the 3D electrode interface design mode of the tool that allows designers to design custom electrode arrays on 3D surfaces, converting them to customised electro-tactile interfaces. Even though similar work has been done by Groeger et al. [8], there remains a requirement for tools that do not require closed-source software and also provide completion for the design pipeline of eTactileKit. Designers can import their 3D models as *.stl* or *.obj* files and reduces the mesh to allow for ease of mesh manipulation operations in the tool. The size and spacing of the electrodes and connectors can be defined. The designer can select faces on the 3D model that are intended to be electrode regions, and the tool will automatically populate the electrode arrays on the selected regions with approximately uniform distances as shown in Figure 5-b. Similarly, connector nodes can be populated. We use Python's *Pyvista* and *PyACVD* libraries to handle the mesh operations within the tool. The approximate equidistant population of electrodes and connectors is done by subdividing the selected region to a finer mesh and then selecting a set of vertices such that they approximate an equidistant set of vertices. As the next step, the tool routes the connections between the electrodes and connectors using the A* algorithm (see Figure 5-c). Finally, the 3D routing traces and mesh objects can be exported separately as *.obj* files that can be sliced and 3D printed. To accomplish the desired outcomes, multi-material 3D printing where the original mesh is printed with non-conductive materials and the 3D routing traces with conductive materials [1, 28, 36] can be used [9]. Moreover, the tool exports a JSON file containing the electrode locations to interface the electrode designs with the subsequent software.

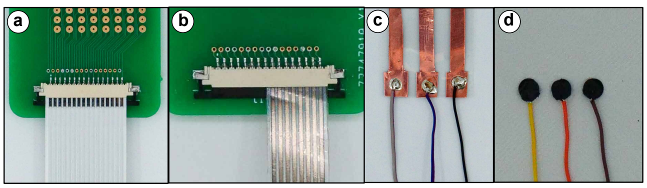

Figure 6: Connecting electrode interfaces to the switching module using: (a) a standard FPC cable, (b) an inkjet printed traces designed to fit the connector pitch, (c) manually cut copper tape traces soldered to wires, and (d) wires attached to 3D printed connectors by heat bonding.

4.2.3 Electrode and Hardware Interfacing. We use several methods to connect fabricated electrode interfaces and our switching module. For the PCB electrodes, we use FPC cables with 1mm pitch (see Figure 6(a)). The number of circuits (conductive paths) in the FPC cable can be no more than 16 to fit the connector. For fabrication methods such as inkjet printing, the connector pads of the electrode layout can be directly designed and connected to the switching board as shown in Figure 6(b). For all the other non-standard connector approaches, as shown in Figure 6(c) and (d), we use an Adafruit FPC Stick (1mm pitch) to extend connections to the switching board.

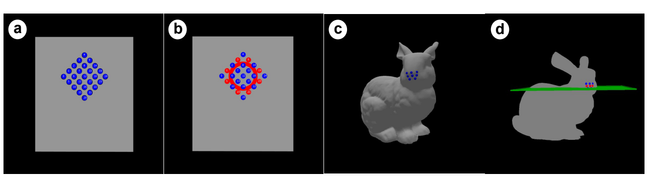

Figure 7: eTactileKit's pattern creator tool: (a) the visualizer showing the imported 2D electrode array and shape outline of a tactile display, and (b) simulation of a diverging ring activation pattern. (c) Similarly, a 3D electrode array is visualised on the Stanford Bunny's nose, and (d) are activated when overlapping with a moving plane.

```
ETactile eTactile;

// Pattern variables
float centerX       = 300;
float centerY       = 300;
int   radius        = 5;
int   stroke_width  = 10;

String electrode_board_file  = "electrode_array_main_board.json";
String output_file           = "diverging_ring_pattern";

void setup() {
  size(1000, 900);
  eTactile = new ETactile(this);

  eTactile.setup2D(electrode_board_file, output_file);
}

void draw() {
  stroke(255, 0, 0);
  strokeWeight(stroke_width);
  noFill();

  ellipse(centerX, centerY, radius * 2, radius * 2);
  radius += 5;

  eTactile.eTframedelay(100);     // delay between two frames
  eTactile.eTfreq(75);            // stimulation frequency
  eTactile.eTmode("Anodic");      // stimulation mode

  if (radius >= 250) {
    eTactile.terminate();
  }
}
```

Figure 8: An instance of eTactileKit's pattern creator interface used to create a diverging ring-shaped pattern.

4.3 Stimulation Pattern Creator (R2)

Stimulation patterns are spatially, temporally and sometimes even parametrically defined activations of electrodes. Creating stimulation patterns by controlling each individual electrode can be a cumbersome process when the number of electrodes scales (i.e., 8×8 array has 64 individually controllable electrodes). This is further complicated by other factors such as frequency and stimulation mode. Moreover, novel electro-tactile sensations can be explored with accurately controlled spatiotemporal stimulation patterns [11, 48], underscoring the need for complex control mechanisms. Thus, a software tool to assist designers in easily designing stimulation patterns is paramount for the toolkit.

Previous haptic toolkits have used GUI-based approaches to create stimulation patterns [26, 42]. However, these approaches are less flexible for complex stimulation patterns and limit scalability when the number of electrodes increases. To avoid such complications in GUI-based approaches, we drew inspiration from a simplified scripting platform *Processing* developed for graphics, to design the stimulation pattern creator in eTactileKit. Furthermore, when designing tactile experiences, designers often need to iterate rapidly between designing and testing the stimulation patterns. Thus, supporting visual simulation of stimulation patterns helps designers in the design process for quick iteration and debugging.

Therefore, eTactileKit offers a simplified scripting and visualisation interface via a *Processing* library module to program tactile stimulation patterns and visualise the simulation (R2a, R2b). To simplify the explanation, we use an example of creating a stimulation pattern resembling a diverging ring on the electrode array as shown in the *Processing* sketch in (Figure 8). The *Processing* library `ETactile` is built, encapsulating all the electrode-wise handling tasks, allowing the designers to focus solely on generating graphics on the electrode canvas. The overlapping electrodes with the simulated graphics (e.g., the ring Figure 7-b) will be marked as active by the library, internally creating individual electrode activation timing and duration. This does software-level abstraction, hindering the underlying complexity in the ground-up programming of electro-tactile stimulation patterns. Software design patterns [2], such as visual feedback and modular architecture, were used to improve usability and scalability.

In the example sketch, the variable `electrode_board_file` refers to the electrode layout exported from the electrode interface design tool. `output_file` defines the file name to record the created stimulation pattern. When `ETactile` class is instantiated within the `void setup()`, it visualises the electrodes and the outline as shown in Figure 7-a and Figure 7-c. In the 3D electrode scenario, both the original mesh object and the electrode layout need to be imported. We use the *PeasyCam* library for 3D object visualisation.

Next, the center coordinates (`centerX, centerY`), the initial ring radius *(radius)*, and the stroke width of the ring *(stroke_width)* are defined. The `void pattern()` function, which is called at each frame by default, includes the graphics element and how it should change from one instance to another as programmed by the designers. In this example scenario, a ring is created with a defined stroke width of 10 units, and the radius is incremented by 5 units in each frame. The colour of the drawn ring is set to red using the `stroke(255, 0, 0)`, where `(255, 0, 0)` represents RGB intensities. `eTframedelay(100)` is used to set the delay between two frames to 100ms. This creates a diverging ring pattern on the electrode canvas (see Figure 7-b). The library handles electrode activations based on overlaps with the created ring. Simultaneously, it updates the visualisation window and stores the activated electrodes in each frame. For the 3D scenario, a function to calculate the inclusion criteria for the electrodes based on the created 3D graphic element needs to be created. `terminate()` stops the pattern generation when the radius exceeds 250 units. Here, the stimulation patterns will be saved as a JSON file with the output file name.

While the sketch shown in Figure 8 is a simple example, designers can create complex graphics on the canvas using other libraries available in Processing. Furthermore, there are other electro-tactile parameters such as frequency (`eTfreq`), stimulation mode (`eTmode`) that can be programmed to explore different tactile sensations. For more details on the available parameters and their usage, we encourage readers to refer to the eTactileKit documentation.

4.3.1 *Pattern Library (R2c).*

To allow designers to experience stimulation patterns quickly and assist novice designers in programming patterns, the tool provides a set of predefined stimulation patterns in a library. This is similar to the examples available in Arduino IDE, such as blinking, fading, smoothing, etc. These patterns include diverging ring activation, blinking, horizontal/vertical scanning and moving planes for 3D. This allows designers to import basic building blocks of stimulation patterns and modify them if needed to create more complex patterns. Furthermore, this library can be extended by adding custom patterns, also allowing designers to create and contribute back to the library for future iterations.

Figure 9: eTactileKit's interaction tool: (a) A 2D and (b) a 3D interface loaded to the GUI, and stimulation patterns are being executed.

4.4 eTactileKit Interaction Tool (R2)

Prior to executing the stimulation patterns on hardware, it is vital to ensure that the stimulation intensity is properly calibrated to the user's comfort level. The formative study revealed that this was mostly done using trial and error methods, which can lead to discomfort and raise safety concerns amongst designers. Moreover, it revealed that the designers explore various stimulation parameters hardcoded into the firmware, losing access to dynamically control them, which requires iterative firmware flashing for updating.

To address these, eTactileKit provides an interactive tool with a GUI that provides a streamlined process to calibrate, execute the created patterns and explore various electro-tactile-based experiences in real-time. To get started, the designer imports both the designed electrode layouts and the stimulation patterns created in the previous steps to the GUI.

4.4.1 *Inbuilt Fast Calibration (R2d).*

The software allows the calibration of the intensity of the stimulation patterns to two thresholds: the lower threshold and the upper threshold. This ensures a safer electro-tactile experience. The lower threshold is the starting point of electro-tactile perception when the intensity increases from 0, and the upper threshold is the starting point of pain. The tool assists designers to gradually increment intensity and record the lower threshold and the upper threshold, subjectively. These parameters are saved and raise warnings if exceeded during interaction.

4.4.2 *Realtime Exploration (R2e).*

Tactile perceptions are often not solely reliant upon spatiotemporal activation of electrodes, but also on other stimulation pulse parameters [17, 51]. To satisfy this requirement, the tool allows easy manipulation of pulse parameters in real time using a set of UI elements. These parameters include stimulation pulse width/height, sense pulse width/height, stimulation frequency, stimulation mode (anodic/cathodic), and channel discharge time. This allows designers to explore various electro-tactile-based experiences and sensations. These parameters were explained further in Section 4.1.

To expand the scope of real-time exploration and add a method to quickly validate electrical connections and functionality of each electrode, the tool consists of a feature to preferentially activate individual electrodes (R2f). This mode allows designers to click an electrode on the UI and feel the electro-tactile sensation from that electrode by activating it.

Moreover, to understand the current activation states of the electrodes, the GUI provides a visual representation of the activated electrodes in real-time, as shown in Figure 9-(a, b). Additionally, to understand the impedance across the electrodes and the skin, the GUI displays the ADC readings corresponding to each electrode. This further allows for gaining insights into the contact impedance with the skin and the electrodes. This is achieved by inbuilt sensing circuitry explained in Section 4.1.1.

4.5 eTactileKit APIs (R3f)

Electro-tactile technology is often not limited to isolated tactile feedback applications but has been used in a wider spectrum of applications, including but not limited to VR [52], mobile platforms [47] and co-functioning with mechanical actuators [45]. Thus, to broaden the applicability and extensibility of eTactileKit to different technologies, it provides APIs for Python and Unity. APIs can bypass the Pattern Creator and GUI software and directly interact with the hardware with full control over the stimulation parameters. The API encapsulates the complexity of the hardware-software interfacing and provides designers with the ability to program at a higher level. Moreover, we are currently developing APIs for Android mobile platforms and enabling wireless access, which we are planning to release in future iterations and welcome the community to contribute constructively to the toolkit.

4.6 Comprehensive Documentation (R4)

To overcome the lack of accessible knowledge sources and conceptual understanding about electro-tactile, eTactileKit provides comprehensive documentation with a complete set of steps and guidelines to use each component of eTactileKit (R4b), using curated examples. Visual aids, such as GIFs and images, are used to improve clarity. It is documented to help both novices and experienced designers, including quick start guides (R4a) to deeper technical details on the eTactileKit. A set of safety guidelines is included to ensure secure operation, addressing key user safety considerations that emerged from the formative study (R4c). The documentation is available in the GitHub repository[1].

5 Applications

To demonstrate the potential of eTactileKit, we developed 4 example applications using eTactileKit.

5.1 Interacting with Virtual Objects in VR

To demonstrate the toolkit's ability to extend to commonly used platforms like Unity and to emphasise the API's ability to bypass the interaction tool, we developed a finger-worn electrode interface (see Figure 10-a1) that enhances user interaction with objects in virtual reality (VR). We used the Unity API to load preset stimulation patterns created by the pattern creator and activate these patterns when pressing (see Figure 10-a3) and releasing (see Figure 10-a2) different buttons in a virtual interface. These tactile profiles can resemble the sensation of pressing a soft profile while another simulates the firm click of a mechanical switch. Moreover, this is an inkjet-printed electrode interface showing the applicability of rapidly fabricated interfaces.

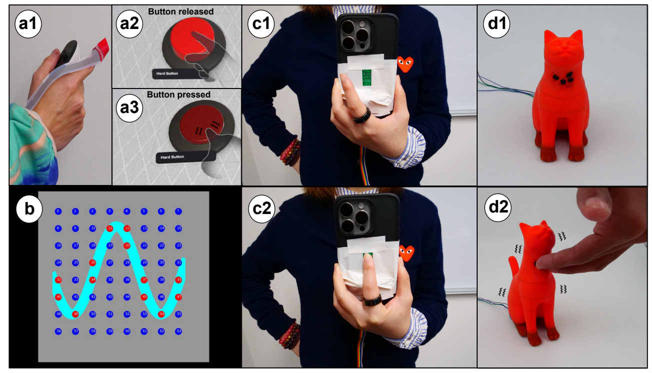

Figure 10: Application scenarios for eTactileKit. (a1) A user wearing a finger-worn electrode interface on the right index finger (a3) pressing and (a2) releasing a virtual button. (b) A moving sinusoidal GIF is loaded onto the electrode canvas of the stimulation pattern creator. (c1)-(c2) An add-on widget for mobile devices to get notifications and give user inputs. (d1)-(d2) An interactive 3D cat toy that delivers tactile feedback mimicking purring when the user interacts with it.

5.2 Extended Stimulation Pattern Creator

In order to highlight the toolkit's ability to be customised, we extended the stimulation pattern creator to support importing GIF elements to the electrode canvas to create stimulation patterns. This eliminates the requirement for scripting and allows the creation of stimulation patterns by simply importing a GIF. Moreover, this creates the possibility to generate more intricate shapes other than the generic ones generated by scripting. Figure 10-b shows a moving sine-wave GIF loaded to the electrode canvas, and the overlapping electrodes are activated simultaneously.

5.3 An Add-on Widget for Mobile Devices

To show the potential of eTactileKit augmenting existing devices, we developed a compact electro-tactile add-on widget that can be attached to the back of a mobile phone as depicted in Figure 10-(c1, c2). This PCB widget enhances user and device interaction by delivering private tactile notifications rather than relying solely on vibrations or sound. Beyond notifications, for accessibility, it could provide alternative feedback for visually impaired users, offering tactile cues for navigation within apps or the physical world.

5.4 Adding Interaction to 3D Printed Objects

Interactive toys that provide sensory feedback have been shown to offer therapeutic benefits for individuals with autism, particularly in regulating sensory input and promoting emotional engagement [21, 38]. In this scenario, we used the 3D electrode interface development tool to create a 3D-printed toy cat that reacts to user interactions through electro-tactile feedback, as shown in Figure 10-d1. This interactive model mimics purring when gently stroked in the neck region (see Figure 10-d2), providing a lifelike response that enhances the sensory experience. Moreover, this application demonstrates the potential of combining 3D printing with electro-tactile technology.

6 Evaluation

To evaluate eTactileKit, we conducted a longitudinal study to assess (1) the usability of the toolkit, (2) how it lowers the barrier to entry for designing and prototyping electro-tactile applications for novice designers, and (3) how the toolkit affects the design workflow.

6.1 Procedure

Our evaluation approach follows similar approaches taken in previous work [7, 22, 24, 27], where they conducted take-home studies within a given period of time. The study spanned over 3 weeks, during which the participants developed applications and used the toolkit for their electro-tactile-related projects. All resources were provided to participants before the study began to set up. The participants were briefly introduced to the toolkit and its components on the first day. The participants maintained a diary of their experiences with the toolkit, challenges faced, and suggestions for improvement of the toolkit on a daily basis for the days they used the toolkit. We conducted internet interviews with open-ended questions to gather feedback after the 1st and 3rd weeks of the study. These were recorded and transcribed for further analysis. Two authors coded the transcriptions to identify the key themes and insights from the participants. No major technical issues with the toolkit components were reported during the study.

6.2 Participants

The participants included 7 males and 1 female with an average age of 25.9 years (SD = 3.6). They were from 5 different labs from 4 different countries. The participants included 4 novice participants (PN1 to PN4), who had no prior experience with electro-tactile and 4 experienced participants (PE5 to PE8). All participants had HCI research backgrounds, which included haptics, software development, and wearable computing development. The novice participants wanted to do research projects related to electro-tactile. They were interested in haptic illusions, electro-tactile feedback in earables, electro-tactile feedback in VR for people with visual impairments, and forehead electro-tactile interfaces, respectively. The experienced participants were involved in investigating the sense of adhesion, assisting navigation for visually impaired people, electro-tactile with force feedback, and using visual cues to alter the subjective pain perception in electro-tactile feedback, respectively.

6.3 Results

Based on the themes and insights from the participants, we present our results in getting started with the toolkit, usability, the effect on the design process of both novice and experienced participants, and the toolkit interaction outcomes.

6.3.1 Getting Started with the Toolkit. The toolkit helped participants get started with electro-tactile stimulation quickly. All participants managed to implement their first stimulation pattern in their first attempt, and four participants reported that they implemented their first stimulation pattern in less than 3 hours. PN1 commented, *"This toolkit actually made me shorten the time of getting to know the electro-tactile... everything is there"*. It provides the resources for end-to-end development, suppressing the need for participants to search for resources (n=4). For instance, PN3 commented, *"The toolkit is a shortcut ... I don't have to learn each individual software that is needed... we can just do the important stuff straight away"*.

All participants highlighted the impact of the documentation. The short videos, along with descriptive guidance, enabled an easy understanding of how each component and step worked while using the toolkit (n=6). PE6 mentioned *"It has videos (GIFs) and each step... it's descriptive. I don't have to look at anything else"*. The participants also appreciated the structuring of the documentation. PN2 mentioned, *"The documentation was very helpful and clear"*.

6.3.2 Ease of Use. Participants only used the components of the toolkit relevant to their project. However, each toolkit component was used by at least one participant. The participants emphasised the structured architecture, making the process easy to understand (n=3). For instance, PE6 mentioned, *"It is all related to each other... it's kind of like a workflow and everything is step by step"* and PN1 mentioned, *"well structured... very clear to understand for the user"*. Moreover, in general, the participants did not find any difficulty in using the components of the toolkit they used and specifically highlighted at least one component as *"easy to use"*. PE8 mentioned, *"I thought it was easier to operate intuitively than general software that should be used [to achieve the same task]"*. Using visual feedback in pattern creation and the GUI enabled the participants to program bug-free patterns and also experience what is happening on hardware in real-time (n=3). For example, PN3 mentioned, *"visualisation is really useful... I can check whether stimulated pattern also matches the created pattern"*. The participants appreciated the calibration procedures, which led to user comfort and safety assurance when using the toolkit (n=2). PN1 mentioned, *"I don't really have to do all the trial and error,... being electrocuted myself"*.

6.3.3 Flexibility and Extensibility. The toolkit's flexibility to adapt to other technologies was highlighted by the participants (n=6). For instance, PN4 mentioned, *"you can change the tactile sensation with just a single line of code [with the Python API]"* and PN1 mentioned, *"I usually do all the haptic research based on Unity. So, it was very helpful for the API to be there"*. The toolkit allowed customisation and scaling for the participants (n=2), marking its extensibility. Since all the source codes were available to the participants, they were able to customise the components to their needs. PN1 mentioned, *"could make my own libraries out of it... I got control of the toolkit"*. Furthermore, the toolkit allowed the hardware to be scaled for more electrodes, underscoring its scalability. PN3 mentioned, *"I can stack and use as much as electrodes as I need"*. Additionally, the toolkit facilitated alternative approaches for the participants. For instance, PN3 mentioned, *"I manually edited a [sample] JSON file instead of using the [stimulation pattern creator] to create the pattern"*, without the need to program the pattern, which opened possibilities for designers novice to programming. However, they acknowledged that this approach would become cumbersome for complex patterns and a larger number of electrodes.

6.3.4 Designer satisfaction. The participants expressed their interest in continuing to use at least one sub-component of the toolkit for their research (n=8), emphasising designer satisfaction. PE5 stated, *"2D Pattern Creator and the Real-Time Hardware Access GUI, which I found to be particularly useful for creating a variety of tactile patterns that fit my research"* and PN2 mentioned, *"I'm planning to use Unity to explore the integration of electro-tactile and music"*.

6.3.5 Effect on Design Process. The toolkit enabled quick prototyping of electro-tactile applications (n=5). PN2 stated the advantage of having onboard electrodes to *"quickly test a pattern without connecting to other boards [electrode interfaces]"*. Specifically, the pattern creator and the GUI allowed rapid design iteration. For instance, PE7 mentioned, *"The GUI was useful for quickly trying out patterns"*. The simplified programming interface of the pattern creator allowed the designers to focus on the pattern itself without focusing on the underlying complexities of pattern creation. For example, PN2 stated, *"not much coding is involved, which is time saving"*. Moreover, the GUI allowed the exploration of sensations in real time by tweaking pulse parameters. PN4 mentioned, *"It was exciting to directly experience how changes in those parameters affect the haptic sensations"*.

The experienced participants (2 out of 4) clearly found that having the toolkit earlier would have enabled them to focus more on design exploration with electro-tactile. PE6 explicitly mentioned, *"compare it with my previous experience… it is really easy to learn how to implement patterns and how to do… PCB board design… it would have saved a lot of time"*. PE5 stated *"if I created that stimulation with this toolkit, the amount of control in Processing would be reduced"*. Moreover, they experienced new parameters and design possibilities with the toolkit. For instance, PE7 mentioned, *"electrical stimulation pattern can be changed more flexibly"*, and PE5 mentioned, *"the ability to visually change the stimulation frequency and discharge time in real-time is something I have not seen before"*.

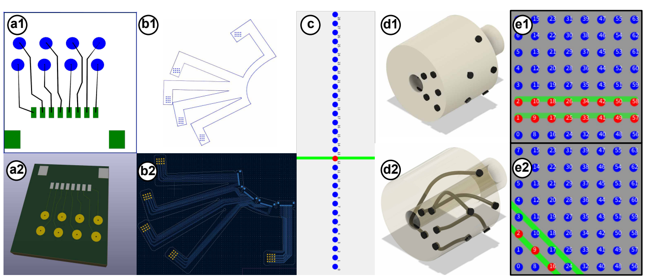

Figure 11: Applications developed by participants using eTactileKit. (a1) Layout view on the tool and (a2) 3D view of a PCB desinged. (b) A Flexible PCB layout designed to fit the fingertips of the hand, designed for VR interaction. (c) A 1D electrode array was created to explore sinusoidal activation patterns. (d) Electrodes placed on an ear tip for in-ear electro-tactile feedback. Experimental patterns (e1) vertical scanning and (e2) diagonal scanning patterns.

6.3.6 Toolkit Interaction Outcomes. The participants were asked to share the outcomes of the work they had been doing if that did not affect their research privacy. PN4 shared a basic electrode layout (see Figure 11-(a1, a2)) designed using the electrode design tool to get started. PN3 shared the flexible PCB layouts designed to fit the fingertips shown in Figure 11-(b1, b2), consisting of 64 electrodes per hand, showing the tool's potential to support complicated electrode adjustments and routing. PN2 explored a sinusoidal stimulation pattern on a 1D electrode array (see Figure 11-c), exploring the possibilities of creating electro-tactile feedback based on audio data. Furthermore, PN2 shared a customised ear-tip design with electrodes placed on it for electro-tactile feedback (see Figure 11-(d1, d2)). PE6 created a layout for an existing electrode PCB to create stimulation patterns to assist navigation for visually impaired people, with electrodes attached on mobile devices (see Figure 11-(e1, e2)). These patterns were executed using the interaction tool.

7 Limitations and Future Work

7.1 Diversity of Applications

Electro-tactile stimulation is gaining popularity and is being applied in various domains, as highlighted in Section 2.1. As research continues to explore these emerging frontiers, there is a growing demand for versatile, easily integrated toolkits that can accommodate different development environments and hardware configurations.

The version of the toolkit provides APIs for Python and Unity, which are easy to adopt and widely used in the community. However, we believe that the toolkit must be extended to support web technologies (e.g., NextJS) and mobile frameworks (e.g., Android). This will enhance the accessibility of the toolkit and allow a wider range of developers to utilise the electro-tactile stimulation technology in their applications.

The current hardware setup is designed to be modularised and preserves its DIY nature, resulting in a higher form factor. However, re-engineering hardware with specialised circuits or form factors can open electro-tactile stimulation to embedded and wearable devices where size, ergonomics, and power consumption are critical. For example, prior works have developed smaller and application-specific hardware form factors [47]. Since we open-source all resources, we believe designers can develop customised hardware by limiting redundant components (switching ICs), enabling smaller form factors suitable for applications such as wearable devices.

Even though our hardware implementation follows a layered safety architecture, we recognise that the possibility of hardware failure cannot be completely avoided. Therefore, re-engineered hardware can benefit from additional safety protocols in future iterations. Moreover, designers should take careful notice of possible direct high-voltage contact during hardware debugging.

7.2 Integrating Future Advancements

While eTactileKit's existing features mark a strong starting point, as shown by the evaluation results, the current version can be considered a beta release. We believe its approaches and methods will benefit from ongoing community collaboration and collective improvements, especially as new technological requirements emerge.

For instance, the electrode interface design tool provides a single-layer routing solution. However, designers can still make modifications to the PCBs in KiCAD for more complex designs. Enabling multi-layer routing in the tool will be a good feature for future iterations that allows the design of more complex electrode layouts.

Similarly, automatic routing algorithms used in the tool will not always provide the best routing solution and routing for all the connections. Still, it will provide a good routing solution for most connections. In the 2D electrode interface design tool, manual routing will allow routing of any missed connections. More advanced routing algorithms can be implemented to provide better routing solutions. In the 3D scenario, the tool can be modified to adopt more powerful automatic routing solutions, as shown in prior works [30].

Furthermore, the calibration procedure helps to identify the desirable stimulation current for each user, ensuring a nicer experience with electro-tactile. The current calibration procedure requires manual adjustment of the stimulation parameters for the lower and upper thresholds for just noticeable sensation and pain sensation, respectively. However, there is a considerable spatial variation of the aforementioned threshold values across the skin. [13]. To overcome this, future work should consider more advanced calibration procedures that can be implemented to automatically adjust the stimulation parameters based on the user's skin impedance, following the techniques used by prior works [13, 34].

7.3 Participant Suggestions

While many participants from the longitudinal evaluation emphasised its comprehensiveness and usability, their experiences also illuminated areas where further enhancements could improve workflow and accessibility, especially for those transitioning from preexisting design methods.

For instance, experienced participants mentioned that they were accustomed to using the previous methods and had to adapt to the new toolkit. For instance, with prior experience using PCB design tools (n=2), they mentioned that they would have more control and easily *design the board [electrodes] directly in KiCad*. However, they appreciated other components of the toolkit. They brought up the point that it would be an advantage to have the option to export the electrode layout from KiCAD to the toolkit in future iterations.

Moreover, the toolkit simplifies the approach but still requires basic programming knowledge to create customised stimulation patterns using the stimulation pattern creation tool (n=3). For instance, PN1, PN2, and PN3 suggested *"GUI"* for pattern creation would allow complete novices to enter into electro-tactile and when engaging with small-scale electrode designs. To address this, we developed an extended version of the pattern creation tool that allows designers to import GIFs, negating the need for programming knowledge. This feature was unavailable during the longitudinal evaluation but was developed based on participant feedback. We believe that the extensibility of the toolkit, as aforementioned, will enable customisation of the toolkit to their needs and further enhance the designer experience.

8 Conclusion

We presented eTactileKit, an electro-tactile toolkit that enables the design exploration and rapid prototyping of electro-tactile interfaces. It offers hardware and software support at all stages for creating electro-tactile interfaces. Results from our longitudinal evaluation showed that eTactileKit is (1) easy to use, (2) lowers the barrier to entry for designing and prototyping electro-tactile applications, and (3) even enhances the current workflow of experienced designers. A set of example applications developed along with outcomes from designer interactions from the longitudinal study demonstrated the potential of eTactileKit in enabling designers to create novel electro-tactile interfaces that enhance user experiences in various domains. We believe that eTactileKit will empower designers to explore the design space of electro-tactile interfaces and inspire new applications that leverage tactile feedback in interactive systems. We further believe all the open-source resources will inspire future work to consider miniaturising the hardware, seamlessly integrating with wearables and smaller devices, exploring novel tactile sensation spaces, and integrating new sensing modalities such as capacitive sensing to create more expressive interfaces.

Acknowledgments

This project was funded by the Australian Research Council Discovery Early Career Award (DECRA) - DE200100479. Dr. Withana is the recipient of a DECRA funded by the Australian Government. This work was also partly supported by the Deutsche Forschungsgemeinschaft (DFG) under Grant No. 521601028. We also thank all user study participants for their valuable time. Additionally, we appreciate the members of the AID-LAB for their assistance.

References

[1] Multi 3D. [n. d.]. Electrifi Conductive Filament. Retrieved 25/03/2025 from https://www.multi3dllc.com/product/electrifi/
[2] Boyan Bontchev and Emanuela Milanova. 2020. On the Usability of Object-Oriented Design Patterns for a Better Software Quality. *Cybernetics and Information Technologies* 20, 4 (2020), 36–54. doi:10.2478/cait-2020-0046
[3] Victoria Clarke and Virginia Braun. 2021. Thematic analysis: a practical guide. (2021). https://www.torrossa.com/it/resources/an/5282292
[4] Marta Franceschi, Lucia Seminara, Strahinja Dosen, Matija Strbac, Maurizio Valle, and Dario Farina. 2017. A System for Electrotactile Feedback Using Electronic Skin and Flexible Matrix Electrodes: Experimental Evaluation. *IEEE Transactions on Haptics* 10, 2 (2017), 162–172. doi:10.1109/TOH.2016.2618377
[5] Michele Germani, Maura Mengoni, and Margherita Peruzzini. 2013. Electro-tactile device for material texture simulation. *The International Journal of Advanced Manufacturing Technology* 68 (2013), 2185–2203. https://doi.org/10.1007/s00170-013-4832-1
[6] Saul Greenberg. 2007. Toolkits and interface creativity. *Multimedia Tools and Applications* 32 (2007), 139–159. doi:10.1007/s11042-006-0062-y
[7] Saul Greenberg and Chester Fitchett. 2001. Phidgets: easy development of physical interfaces through physical widgets. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology* (Orlando, Florida) *(UIST '01)*. Association for Computing Machinery, New York, NY, USA, 209–218. doi:10.1145/502348.502388
[8] Daniel Groeger, Martin Feick, Anusha Withana, and Jürgen Steimle. 2019. Tactlets: Adding Tactile Feedback to 3D Objects Using Custom Printed Controls. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 923–936. doi:10.1145/3332165.3347937
[9] Liang He, Jarrid A. Wittkopf, Ji Won Jun, Kris Erickson, and Rafael Tico Ballagas. 2022. ModElec: A Design Tool for Prototyping Physical Computing Devices Using Conductive 3D Printing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 4, Article 159 (dec 2022), 20 pages. doi:10.1145/3495000
[10] Atsuki Higashiyama and Mamoru Hayashi. 1993. Localization of electrocutaneous stimuli on the fingers and forearm: effects of electrode configuration and body axis. *Perception & psychophysics* 54 (1993), 108–120. https://link.springer.com/content/pdf/10.3758/BF03206942.pdf
[11] Arata Jingu, Nihar Sabnis, Paul Strohmeier, and Jürgen Steimle. 2024. Shaping Compliance: Inducing Haptic Illusion of Compliance in Different Shapes with Electrotactile Grains. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 432, 13 pages. doi:10.1145/3613904.3641907
[12] Arata Jingu, Anusha Withana, and Jürgen Steimle. 2023. Double-Sided Tactile Interactions for Grasping in Virtual Reality. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 9, 11 pages. doi:10.1145/3586183.3606798

[13] Hiroyuki Kajimoto. 2011. Electrotactile display with real-time impedance feedback using pulse width modulation. *IEEE Transactions on Haptics* 5, 2 (2011), 184–188. doi:10.1109/TOH.2011.39

[14] Hiroyuki Kajimoto. 2016. *Electro-tactile Display: Principle and Hardware*. Springer Japan, Tokyo, 79–96. doi:10.1007/978-4-431-55772-2_5

[15] Hiroyuki Kajimoto. 2021. Electro-Tactile Display Kit for Fingertip. In *2021 IEEE World Haptics Conference (WHC)*. 587–587. doi:10.1109/WHC49131.2021.9517192

[16] Hiroyuki Kajimoto, Yonezo Kanno, and Susumu Tachi. 2006. Forehead electro-tactile display for vision substitution. In *Proc. EuroHaptics*, Vol. 11. Citeseer. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=4d61efe8fd515c55fb6f8c3f5c5bf8d48cffc2b4

[17] Hiroyuki Kajimoto, Naoki Kawakami, T Maeda, and S Tachi. 2004. Electro-tactile display with tactile primary color approach. *Graduate School of Information and Technology, The University of Tokyo* (2004). http://www.files.tachilab.org/publications/intconf2000/kajimoto200410IROS.pdf

[18] H. Kajimoto, N. Kawakami, S. Tachi, and M. Inami. 2004. SmartTouch: electric skin to touch the untouchable. *IEEE Computer Graphics and Applications* 24, 1 (2004), 36–43. doi:10.1109/MCG.2004.1255807

[19] Hiroyuki Kajimoto, Masaki Suzuki, and Yonezo Kanno. 2014. HamsaTouch: tactile vision substitution with smartphone and electro-tactile display. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI EA '14)*. Association for Computing Machinery, New York, NY, USA, 1273–1278. doi:10.1145/2559206.2581164

[20] Kunihiro Kato, Hiroki Ishizuka, Hiroyuki Kajimoto, and Homei Miyashita. 2018. Double-sided Printed Tactile Display with Electro Stimuli and Electrostatic Forces and its Assessment. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. doi:10.1145/3173574.3174024

[21] Kesavan Krishnan, Nazean Jomhari, Ramesh Kumar Ayyasamy, Sameem Abdul Kareem, and Sathis Krishnan. 2024. Haptic Feedback: An Experimental Evaluation of Vibrations as Tactile Sense in Autistic People. *IEEE Access* (2024). doi:10.1109/ACCESS.2024.3410845

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

[23] Hirose Electric Co. Ltd. [n. d.]. FH12-16S-1SH(55). Retrieved 25/03/2025 from https://www.hirose.com/en/product/search/?q=FH12-16S-1SH(55):name-asc:productType:product:terminationProduct:01_SALE:terminationProduct:02_NRND:terminationProduct:03_S2BD

[24] Nicolai Marquardt, Robert Diaz-Marino, Sebastian Boring, and Saul Greenberg. 2011. The proximity toolkit: prototyping proxemic interactions in ubiquitous computing ecologies. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 315–326. doi:10.1145/2047196.2047238

[25] Microchip. 2019. Microchip HV513. Retrieved 25/03/2025 from https://www.microchip.com/en-us/product/HV513

[26] Dinmukhammed Mukashev, Nimesha Ranasinghe, and Aditya Shekhar Nittala. 2023. TactTongue: Prototyping ElectroTactile Stimulations on the Tongue. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 105, 14 pages. doi:10.1145/3586183.3606829

[27] Michael Nebeling and Moira Norrie. 2012. jQMultiTouch: lightweight toolkit and development framework for multi-touch/multi-device web interfaces. In *Proceedings of the 4th ACM SIGCHI Symposium on Engineering Interactive Computing Systems* (Copenhagen, Denmark) *(EICS '12)*. Association for Computing Machinery, New York, NY, USA, 61–70. doi:10.1145/2305484.2305497

[28] Ninjatek. [n. d.]. EEL 3D Printer Filament (90A). Retrieved 25/03/2025 from https://ninjatek.com/shop/eel/

[29] Antti Oulasvirta and Kasper Hornbæk. 2016. HCI Research as Problem-Solving. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 4956–4967. doi:10.1145/2858036.2858283

[30] Gianpaolo Palma, Narges Pourjafarian, Jürgen Steimle, and Paolo Cignoni. 2024. Capacitive Touch Sensing on General 3D Surfaces. *ACM Trans. Graph.* 43, 4, Article 103 (July 2024), 20 pages. doi:10.1145/3658185

[31] Daniel Pamungkas and Koren Ward. 2015. Immersive teleoperation of a robot arm using electro-tactile feedback. In *2015 6th International Conference on Automation, Robotics and Applications (ICARA)*. 300–305. doi:10.1109/ICARA.2015.7081164

[32] Daniel Sutopo Pamungkas and Koren Ward. 2016. Electro-tactile feedback system to enhance virtual reality experience. *International Journal of Computer Theory and Engineering* 8, 6 (2016), 465–470. https://www.ijcte.org/vol8/1090-E007.pdf

[33] Praneeth Bimsara Perera, Hansa Marasinghe, Taiki Takami, Hiroyuki Kajimoto, and Anusha Withana. 2024. Integrating Force Sensing with Electro-Tactile Feedback in 3D Printed Haptic Interfaces *(ISWC '24)*. Association for Computing Machinery, New York, NY, USA, 48–54. doi:10.1145/3675095.3676612

[34] C.J. Poletto and C.L. Van Doren. 2002. Elevating pain thresholds in humans using depolarizing prepulses. *IEEE Transactions on Biomedical Engineering* 49, 10 (2002), 1221–1224. doi:10.1109/TBME.2002.803563

[35] Narjes Pourjafarian, Anusha Withana, Joseph A. Paradiso, and Jürgen Steimle. 2019. Multi-Touch Kit: A Do-It-Yourself Technique for Capacitive Multi-Touch Sensing Using a Commodity Microcontroller. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1071–1083. doi:10.1145/3332165.3347895

[36] Protopasta. [n. d.]. Electrically Conductive Composite PLA. Retrieved 25/03/2025 from https://proto-pasta.com/collections/all/products/conductive-pla?variant=43265594589432

[37] Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. RetroFab: A Design Tool for Retrofitting Physical Interfaces using Actuators, Sensors and 3D Printing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 409–419. doi:10.1145/2858036.2858485

[38] Ben Robins and Kerstin Dautenhahn. 2014. Tactile interactions with a humanoid robot: novel play scenario implementations with children with autism. *International journal of social robotics* 6 (2014), 397–415. https://doi.org/10.1007/s12369-014-0228-0

[39] Taiga Saito, Takumi Hamazaki, Seitaro Kaneko, and Hiroyuki Kajimoto. 2023. Coldness Presentation to Ventral Forearm using Electrical Stimulation with Elastic Gel and Anesthetic Cream. In *Proceedings of the Augmented Humans International Conference 2023* (Glasgow, United Kingdom) *(AHs '23)*. Association for Computing Machinery, New York, NY, USA, 46–54. doi:10.1145/3582700.3582713

[40] Taiga Saito, Jianyao Zhang, Takayuki Kameoka, and Hiroyuki Kajimoto. 2021. Thermal sensation presentation to the forehead using electrical stimulation: comparison with other tactile modalities. In *2021 IEEE World Haptics Conference (WHC)*. 888–893. doi:10.1109/WHC49131.2021.9517195

[41] Valkyrie Savage, Xiaohan Zhang, and Björn Hartmann. 2012. Midas: fabricating custom capacitive touch sensors to prototype interactive objects. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 579–588. doi:10.1145/2380116.2380189

[42] Hasti Seifi, Sean Chew, Antony James Nascè, William Edward Lowther, William Frier, and Kasper Hornbæk. 2023. Feellustrator: A Design Tool for Ultrasound Mid-Air Haptics *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 266, 16 pages. doi:10.1145/3544548.3580728

[43] Hasti Seifi, Matthew Chun, Colin Gallacher, Oliver Schneider, and Karon E. MacLean. 2020. How Do Novice Hapticians Design? A Case Study in Creating Haptic Learning Environments. *IEEE Transactions on Haptics* 13, 4 (2020), 791–805. doi:10.1109/TOH.2020.2968903

[44] Moshe Solomonow, John Lyman, and Amos Freedy. 1977. Electrotactile two-point discrimination as a function of frequency, body site, laterality, and stimulation codes. *Annals of biomedical engineering* 5, 1 (1977), 47–60. doi:10.1007/BF02409338

[45] Yui Suga, Masahiro Miyakami, Izumi Mizoguchi, and Hiroyuki Kajimoto. 2023. 3D Shape Presentation by Combination of Force Feedback and Electro-tactile Stimulation. In *2023 IEEE World Haptics Conference (WHC)*. 361–367. doi:10.1109/WHC56415.2023.10224481

[46] Yui Suga, Masahiro Takeuchi, Satoshi Tanaka, and Hiroyuki Kajimoto. 2023. Softness presentation by combining electro-tactile stimulation and force feedback. *Frontiers in Virtual Reality* 4 (2023), 1133146. doi:10.3389/frvir.2023.1133146

[47] Taiki Takami, Taiga Saito, Takayuki Kameoka, and Hiroyuki Kajimoto. 2023. ExtEdge: Haptic Augmentation of Visual Experiences of a Smartphone by Electro-Tactile Sensation Through the Edges. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction* (Sydney, NSW, Australia) *(SUI '23)*. Association for Computing Machinery, New York, NY, USA, Article 7, 8 pages. doi:10.1145/3607822.3614543

[48] Seiya Takei, Ryo Watanabe, Ryuta Okazaki, Taku Hachisu, and Hiroyuki Kajimoto. 2015. *Presentation of Softness Using Film-Type Electro-Tactile Display and Pressure Distribution Measurement*. Springer Japan, Tokyo, 91–96. doi:10.1007/978-4-431-55690-9_17

[49] Yudai Tanaka, Alan Shen, Andy Kong, and Pedro Lopes. 2023. Full-hand Electro-Tactile Feedback without Obstructing Palmar Side of Hand. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 80, 15 pages. doi:10.1145/3544548.3581382

[50] Shan-Yuan Teng, Aryan Gupta, and Pedro Lopes. 2024. Haptic Permeability: Adding Holes to Tactile Devices Improves Dexterity. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 417, 12 pages. doi:10.1145/3613904.3642156

[51] Keigo Ushiyama and Pedro Lopes. 2023. FeetThrough: Electrotactile Foot Interface that Preserves Real-World Sensations. In *Proceedings of the 36th Annual ACM*

Symposium on User Interface Software and Technology (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 10, 11 pages. doi:10.1145/3586183.3606808

[52] Sebastian Vizcay, Panagiotis Kourtesis, Ferran Argelaguet, Claudio Pacchierotti, and Maud Marchal. 2022. Design and Evaluation of Electrotactile Rendering Effects for Finger-Based Interactions in Virtual Reality. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology* (Tsukuba, Japan) *(VRST '22)*. Association for Computing Machinery, New York, NY, USA, Article 35, 11 pages. doi:10.1145/3562939.3565634

[53] Chiuan Wang, Hsuan-Ming Yeh, Bryan Wang, Te-Yen Wu, Hsin-Ruey Tsai, Rong-Hao Liang, Yi-Ping Hung, and Mike Y. Chen. 2016. CircuitStack: Supporting Rapid Prototyping and Evolution of Electronic Circuits. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 687–695. doi:10.1145/2984511.2984527

[54] Anusha Withana, Daniel Groeger, and Jürgen Steimle. 2018. Tacttoo: A Thin and Feel-Through Tattoo for On-Skin Tactile Output. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 365–378. doi:10.1145/3242587.3242645

[55] Dennis Wittchen, Katta Spiel, Bruno Fruchard, Donald Degraen, Oliver Schneider, Georg Freitag, and Paul Strohmeier. 2022. TactJam: An End-to-End Prototyping Suite for Collaborative Design of On-Body Vibrotactile Feedback. In *Proceedings of the Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction* (Daejeon, Republic of Korea) *(TEI '22)*. Association for Computing Machinery, New York, NY, USA, Article 1, 13 pages. doi:10.1145/3490149.3501307

[56] Kuanming Yao, Jingkun Zhou, Qingyun Huang, Mengge Wu, Chun Ki Yiu, Jian Li, Xingcan Huang, Dengfeng Li, Jingyou Su, Senlin Hou, et al. 2022. Encoding of tactile information in hand via skin-integrated wireless haptic interface. *Nature Machine Intelligence* 4, 10 (2022), 893–903. https://doi.org/10.1038/s42256-022-00543-y

[57] Vibol Yem and Hiroyuki Kajimoto. 2016. Comparative evaluation of tactile sensation by electrical and mechanical stimulation. *IEEE transactions on haptics* 10, 1 (2016), 130–134. doi:10.1109/TOH.2016.2605084

[58] Junyi Zhu, Jackson C Snowden, Joshua Verdejo, Emily Chen, Paul Zhang, Hamid Ghaednia, Joseph H Schwab, and Stefanie Mueller. 2021. EIT-kit: An Electrical Impedance Tomography Toolkit for Health and Motion Sensing. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 400–413. doi:10.1145/3472749.3474758

A Supportive Evidence for Thematic Analysis Results

A.1 Theme 1: High Barrier to Entry

Many participants expressed that their main source of information was either a colleague or a supervising academic. P1 added that *"the manuals and materials provided by Professor [supervisor name]"*, and P3 mentioned *"Professor [supervisor name] had already prepared resources from the beginning"*. This underscores the reliance on personal networks for guidance. This is true for both hardware and software resources. P6 added that *"hardware was available for me, which someone else has designed"*, referring to a colleague from the same research lab. Additionally, research papers were identified as the main source of information. P7 stated that *"mostly I read research papers to gain knowledge"* and P5 added *"main source of knowledge was research papers"*. However, they were often described as complex and difficult to apply in practical contexts. For instance, P7 added *"would be nice to have some documentation of what does what"*, instead of just research papers. P6 clarified that *"research papers on electro-tactile but they don't go to the specific things about hardware or software, described what they have done in an overview, but not the specific details we need if we are developing something by our ourself"*. P1 further added that, even when they get resources from colleagues, *"documentation's completeness is somewhat unclear"*.

This lack of easily accessible and user-friendly resources creates significant hurdles for those interested in exploring electro-tactile technology, particularly for newcomers or those outside specialised academic backgrounds, such as electronics, e.g., *"learning about electrical circuits was a bit challenging"* (P2). The participants commented on the difficulties in getting started and debugging where they could not *"understand what's going on"* (P5) and often compared the challenges to other frameworks *"It was not easy like understanding Arduino"* (P6). The lack of visual output compared to other frameworks is a big issue. For instance, in Arduino, *"an LED blinks and you know whether it works or doesn't work"* (P7), and *"we sometimes find it difficult to understand the state of electrodes at a glance, so having a clear visual representation of electrode status on [an] UI would be necessary"* (P3).

The findings highlight the need for comprehensive, publicly available, and user-friendly resources comparable to other open-source frameworks to lower the entry barriers in this emerging field.

A.2 Theme 2: Enabling Design Exploration

Participants specifically noted the scarcity of accessible tools for developing hardware. For instance, many participants commented on difficulties in designing electrodes, *"there were no customizable electrode pads. I think it takes some time to do that"* (P6), *"trial-and-error process involved in electrode design"* (P5), and specific requirements around *"size of electrodes and their density played crucial roles"* (P2) and a process that use existing knowledge to *"streamline the [design] process for users"* (P2) could benefit the design pipeline. P6 added an *"easy way to design electrode pads myself and get it out very quick"* was necessary for prototyping. Similarly, understanding, developing, and changing control hardware was also mentioned to be cumbersome and time-consuming, *"Learning about electrical circuits was a bit challenging"* (P2), *"understanding the hardware and ensuring safety took some time"* and *"redesigning the switching board every time can be quite cumbersome"* (P4). P7 summarised it as *"it would have been better if you have some device off the shelf that you are pretty confident that works"*.

The lack of software tools and APIs akin to those available for other tactile modalities further exacerbates this issue. This applied to both firmware and software levels. Participants commented that designing a simple tactile stimulation pattern was complicated by many parameters involved. P2 added, *"specifying whether the stimulation was anodic or cathodic, the pulse height, and the number of channels"*, was complicated, and P6 added *"how to change parameters of like intensity, frequency"* was challenging. P1 mentioned that they had to work on the complicated *"SPI communication at the signal line level"*, and it was very cumbersome, particularly for timing the stimulation. P1 added that it would be *"convenient if functions for determining stimulation electrodes, adjusting current pulses, and determining polarity were already fully available from the beginning"*. P7 added, *"If you can provide an interface so that people don't really have to touch the firmware, that's a big win"*.

Another aspect that the participant commented on was the lack of tools for fast and easy explorations. P2 added, from a designer's perspective, *"it would be helpful to have UI elements like channels for electrode placement"* and, *"Clicking on a channel could trigger electrical stimulation, providing a trial version for experimentation"*.

P4 mentioned the need for *"developing a GUI that assists in configuring multiple electrodes and provides feedback on current settings"*. P3 added *"the capability to change individual electrodes with a click, would be convenient"*. This hints at the need for a tool that helps designers to quickly explore the design space.

This gap in available resources significantly hinders the iterative design process, forcing designers to spend a lot of time and effort on the fundamental technology. The findings underscore the need for more comprehensive and accessible tools—both hardware and software—to support the development of electro-tactile technology. Such advancements would enable designers to shift their focus from foundational work to the creative exploration of tactile experiences, fostering innovation in the field.

A.3 Theme 3: Extensibility and Scalability

Participants highlighted the requirement for extensible and scalable hardware and software solutions to fit various design requirements and application complexities. Most software and communication APIs, particularly firmware, are developed with little consideration for scalability. P3 commented that *"optimising communication protocols for increasing numbers of stimulation points would be beneficial as the complexity of applications grows"*. Furthermore, participants highlighted the need for methods to extend and scale electrode interfaces. P4 added that *"redesigning the switching board"* was necessary to increase the number of electrodes, and *"as the number of electrodes increases, it becomes challenging to consolidate them into a single connector"*. P7 added their hardware setup *"had a fixed [electrode] limit. I think around 8"*, and it constrained the ability to scale the experience.

A.4 Theme 4: Safety and Calibration

Participants expressed concerns about safety and calibration in the design and operation of electro-tactile systems. Safety concerns were raised during experimenting with programmed stimulation patterns. P3 stated that *"mistakes in programming could lead to significant electrical currents flowing through the electrodes, causing discomfort or even injury"*. Moreover, safety concerns were raised while handling hardware, P2 mentioned that *"when it comes to 300 volts, there's a general perception of danger or discomfort associated with it"*. This highlights the need for clear guidelines and tools to ensure safety in design and operation.

Calibration of the system is required to achieve comfortable but yet noticeable tactile perception, preventing discomfort to the user. Participants highlighted the need for methods to expedite the calibration process. This was emphasised in P7's statement, *"calibration is definitely a big thing that the toolkit needs to handle. There needs to be some software support"*. Current calibration practices are often time-consuming and complex, hindering efficient experimentation and deployment. The absence of streamlined, reliable calibration tools further complicates the design process, making it challenging to achieve consistent and safe results. P1 mentioned, *"It seemed that individual differences significantly affected whether the sensation was perceived or not, likely due to differences in stability of contact"*, highlighting the requirement for dynamic adjustment of calibration parameters, which in the absence would lead to inconsistent performance and unreliability. These findings suggest that developing standardised safety protocols and faster, more user-friendly calibration methods would provide much-needed support and confidence for designers working with electro-tactile technology.

B Technical Capabilities of Prior Electro-Tactile Research

We summarise the technical capabilities of prior electro-tactile research and eTactileKit in Table 2.

Table 2: Comparison of eTactileKit and previous electro-tactile systems and toolkits.

System/Toolkit	Pulse Width	Pulse Height	Current Step Size	Max Voltage	Applied Body Region
Electro-Tactile Display: Principle and Hardware [14]	50–200 μs	1–10 mA	–	350 V	Fingertip
Tacttoo [54]	200 μs	0–3.3 mA	16.5 μA	250 V	Fingertip
TactTongue [26]	1–10 μs	20–100 μA	–	–	Tongue
Encoding of tactile information in hand via skin-integrated wireless haptic interface [56]	-	0–13.5 mA	<9 μA	90–135 V	Palm
Tactlets [8]	1–5 ms	0–3 mA	15 μA	400 V	Hand
FeetThrough [51]	250 ms, 100 ms	5.0 ± 1.7 mA, 8.6 ± 5.5 mA	–	–	Foot
Electro-Tactile Feedback System to Enhance Virtual Reality Experience [32]	10–100 μs	–	–	40–80 V	Finger
Full-hand Electro-Tactile Feedback without Obstructing Palmar Side of Hand [49]	50 ms, 400 μs, 800 μs	<4 mA	–	72 V	Hand
Comparative Evaluation of Tactile Sensation by Electrical and Mechanical Stimulation [57]	20–200 μs	–	–	–	Finger
Design and Evaluation of Electrotactile Rendering Effects for Finger-Based Interactions in Virtual Reality [52]	30–500 μs	0.1–9 mA	–	–	Finger
Localization of electrocutaneous stimuli on the fingers and forearm: Effects of electrode configuration and body axis [10]	–	<10 mA	–	–	Finger & Forearm
Shaping Compliance [11]	200 μs	0–2.7 mA	13.5 μA	–	Finger
Double-Sided Tactile Interactions for Grasping in Virtual Reality [12]	200 μs	0–2.7 mA	13.5 μA	–	Finger
Forehead Electro-tactile Display for Vision Substitution [16]	50 μs	–	–	300 V	Forehead
eTactileKit (Ours)	**User defined**	**0–10 mA**	**2.44 μA**	**270 V**	**All regions**