# How Do Novice Hapticians Design? A Case Study in Creating Haptic Learning Environments

Hasti Seifi, *Member, IEEE*, Matthew Chun, Colin Gallacher, Oliver Schneider, *Member, IEEE*, and Karon E. MacLean, *Senior Member, IEEE*

***Abstract*—Access to haptic technology is on the rise, in smartphones, virtual reality gear, and open-source education kits. However, engineers and interaction designers are often inexperienced in designing with haptics, and rarely have tools and guidelines for creating multisensory experiences. To examine the impact of this deficit, we supplied a haptic design kit, custom software, and technical support to nine teams (25 students) for an innovation challenge at a major haptics conference. Teams (predominantly undergraduate engineers with little haptics, interaction design, or education training) designed and built haptic environments to support learning of science topics. Qualitative analysis of surveys, interviews, team blogs, and expert assessments of teams' final demonstrations exposed three themes in these design efforts. 1) Novice teams tended to ignore many of ten *design choices* that experts navigate, such as explicitly choosing whether haptic and graphic feedback should reinforce versus complement one other. 2) Their *design activities* differed in timing and inclusion from the ten activities observed in expert process. 3) We identified three *success strategies* in how teams devised useful and engaging interactions and interpretable multimodal experiences, and communicated about their designs. We compare novice and expert design needs and highlight where future haptic design tools and theory need to support novice practice and training.**

***Index Terms*—haptic design, multisensory interaction, novice designer, haptician, API, education.**

## I. Introduction

*For an engineering course project, Alex and Jing are building a system using an open-source, one-axis, force-feedback device kit provided by the instructor. They decide to create a science-education game. New to haptics and interface design, they brainstorm physics topics and pick one they both found tough in high school: molecular attraction. They choose a 1D game mechanic and some haptic effects.*

*Jing 3D-prints and assembles the device. Alex figures out how to integrate the haptic data stream with an open-source physics engine and a graphics library. With just one device, they must meet in person to test effects. There is lots to figure out as they implement and debug basic behaviors. When they finally get a prototype application working, it does not feel the way they intended, but they are out of time and energy.*

*At the course demo, a guest judge asks them to explain their idea. They tell her about the device, lesson, libraries, and effects. She asks why they think it might work. Alex is puzzled: they just showed a "working" multimodal simulation with haptic effects. The judge explains that she wants to know how this scenario and implementation might help someone learn about molecular attraction. "Hm," says Alex. "There were not a lot of choices once we got the basic game. The hard part was programming the device."*

This scenario, based on observations we report here, typifies our broader experience with novice haptic designers. Hallmarks include reliance on personal experience as inspiration and feedback, a focus on technical factors while neglecting conceptual aspects of the experience, consideration of a narrow part of the available multisensory design space, absence of articulated objectives or criteria, and no plan for iteration. While this process can result in interactions that are delightful and effective, more often it brings disappointment.

Advances in fabrication methods and haptic technology enable new *hapticians* [1] to create diverse and potentially transformative haptic experiences; they may be teachers, therapists, roboticists, application developers or just people. Domains include hands-on learning [2], [3]; improving feedback transparency in robot-assisted surgery [4]; medical training [5], [6]; gaming [7]; physical rehabilitation, and consumer applications requiring end-user customization [8], [9].

*Novice* hapticians (designers new to at least one of haptics or design practice) will need more than improved technology to realize these ideas: obstacles to haptic experience design are legion, even for experienced practitioners. They include the vast diversity of haptic hardware [10], lack of means to describe, sketch, refine, share, and test ideas [11], and a dearth of process examples or guidelines (Fig. 1). To understand their goals, designers need established measures of effectiveness. These vary with design objectives, *e.g.,* from high-fidelity rendering of a surgical training environment (feels like the real

Manuscript received July 25, 2019; revised December 23, 2019; accepted December 30, 2019. Date of publication January 23, 2020; date of current version December 21, 2020. This work was partially funded by the National Science Foundation (USA) and the Natural Sciences and Engineering Research Council of Canada, and enabled by facilities and resources from UBC's Institute for Computing, Information and Cognitive Systems (ICICS) and the UBC Designing for People (DFP) Research Cluster. This article was recommended for publication by Prof. A. M. L. Kappers. *(Corresponding author: Hasti Seifi.)*

H. Seifi, M. Chun, and K. E. MacLean are with the Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada (e-mail: seifi@cs.ubc.ca; mchun345@cs.ubc.ca; maclean@cs.ubc.ca).

C. Gallacher is with the Haply Robotics Co., Montreal, Quebec H2X 2Y7, Canada (e-mail: colin@haply.co).

O. Schneider is with the University of Waterloo, Waterloo, NO N2L 3G1, Canada (e-mail: oliver.schneider@uwaterloo.ca).

This article has supplementary downloadable material available at https://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/TOH.2020.2968903

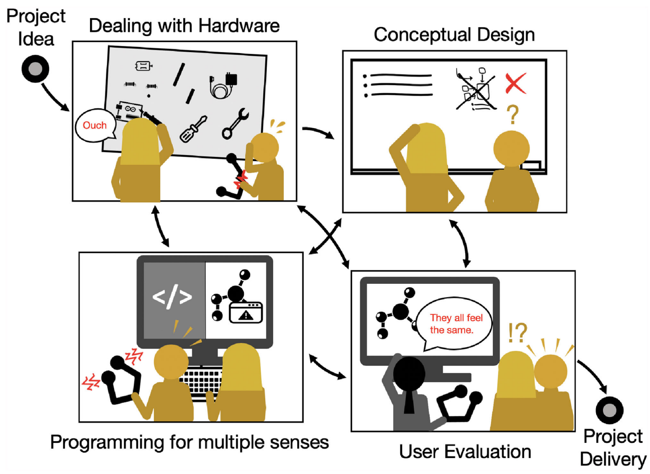



Fig. 1. Haptic design is a messy, complex process. Novice teams need to rapidly switch between diverse activities, from working with hardware and software to conceptual design and evaluation. Novice hapticians, who might be students or domain experts like teachers, need tools and guidelines that support their process.

thing, enhances performance) to embedding haptic elements in the narrative flow of a game or learning experience (understandable, engaging, aids learning, triggers deeper exploration). The few tools that do exist rarely point to a process, and technical challenges can hijack any plan.

While inadequate tools are a problem for everyone, experts are at least better understood. Recent studies have identified important yet poorly supported activities for expert haptic design, such as *browsing* examples and *sketching* new designs [1], [11]. Awareness of practice can inspire tools, and their evaluation for impact on practice [12]–[14]. While research examples are rarely industry-ready, they can lead to specialized in-house tools [15], [16].

Novice hapticians present a useful contrast to experts. Characterizing their challenges gives insight into missing support, including places where experts may be accommodating the tools they have, and leads to usability. Addressing novice needs lowers barriers to entry and teaches novices to design well, creating more capable experts. Supporting tools and guidelines identify different ways of solving problems ("wide walls") while paving the way to "high ceilings" that empower experts [17].

***Approach*.** To learn what *tools, theory, and knowledge infrastructures* are needed to support novice hapticians in designing haptic and multisensory environments, we studied their practice and obstacles, and compared these with expert processes and support documented in haptics and other fields of design. Our findings contribute to an emerging theory of experience design that places haptics in partnership with other sensory modalities [11], and identify opportunities for application design support across a spectrum of competency. We chose science education as a testbed: creators of learning environments are often education technologists or learners themselves, with little haptic or even design experience.

Accessibility of haptic hardware and software (in robustness, simplicity, assembly, and programming) is critical to design and deployment in this context and many others. Low-cost and easy-to-develop-on systems also enable experts in early-design exploration. Thus, we used existing low-cost devices with custom software that prioritized usability, but rather than trying to perfect either one *a priori*, our goal was to identify true accessibility requirements.

Specifically, we conducted a case study of novice hapticians participating in the Student Innovation Challenge (SIC) at the IEEE World Haptics Conference (WHC17), the largest existing meeting on haptics research, in Fürstenfeldbruck, Germany in 2017. We provided 25 students (9 teams from 6 countries) with a design kit of hardware, a software application programming interface (API), documentation, and online technical support (Section III). The challenge was to develop a haptic learning environment for a STEM (science, technology, engineering, math) topic over 9 weeks (IV). We sampled their experiences and results during and at the end of the challenge, via team surveys, a focus group, and input from experts who judged their design outcomes (V). We subjected this quantitative and qualitative data to thematic analysis (VI-VII), and connected our findings to past haptic and design literature (VIII).

We contribute:

1) *Ten parameters*that characterize design choices made by the novice teams, connected to recent literature on haptic and multisensory design parameters (Theme 1).
2) *Identification of ten design activities* teams performed, how well these were supported by the tools we provided, and how they differed from those observed in past work with expert hapticians (Theme 2).
3) *Three success strategies*, synthesized from judge and team comments, that helped teams design engaging and meaningful multisensory interactions (Theme 3).
4) *Implications* for future haptic design tools, theory, and infrastructure required to support hapticians, derived from these themes (Section VIII).

## II. Related Work

Our approach builds on the current state of haptic design theory and tools, and on the value of haptics in STEM learning.

### A. Haptic Design Theory and Expert Practices

***Other design fields have a mature theory.*** As an example, the user experience (UX) community has developed a theory of user interface design, exemplified by Buxton's depiction of an iterative process [18], and well supported by guidelines on the key activities and prototype requirements at each design stage codified in textbooks [19]. A similar rich history exists for product design in the engineering domain [20]. These disciplines have also identified common design pitfalls (*e.g.,* waterfall process; focus on engineering rather than usability), and found approaches to avoid them.

***Haptic experts have adopted non-haptic design practices.*** For example, Haptic Cinematography [21] uses a film-making metaphor, discussing physical effects using cinematographic concepts and establishing principles for editing based on cinematic editing [22]. Similarly, Tactile Animation [13] draws from other audio-visual experiences, and

Cutaneous Grooves [23] draws from music to explore "haptic concerts" and composition as metaphors.

***Emerging haptics-specific design practices face specific challenges.*** Mousette *et al.*'s Simple Haptics paradigm is a way to rapidly prototype haptic hardware and interactions, adapting ideas from the maker movement to the context of interactive touch experiences [24]. Others have sought a *grounded theory* [25] of haptic experience design. Through tools [26], [27] and interviews of their expert haptician users [1], Schneider *et al.* identified four major design activities and associated obstacles at this time:

- *Browsing:* organizing and exploring haptic elements requires representations that align with users' conceptualizations of haptic experiences.
- *Sketching:* sketching tools require new abstractions and representations of haptic sensations and experiences.
- *Refining:* users have to master all components of the haptic pipeline (hardware, firmware, software).
- *Sharing:* haptic media is difficult to share over distances.

We aim to complete this picture by studying novice hapticians and contrasting the results with expert practices.

***Guidance for haptic-specific design choices is sparse.*** All designers need to make decisions, such as when to display what information through which representation or modality (*should I represent attribute A with color or shape?*). Mature design domains such as information visualization define choices, best practices, and roadmaps [28].

Hapticians have generic guidelines which do not address haptic-specific problems [29], as well as literature and courses on haptic perception, control, and device design which do not address the crafting of interactions [30]. Recently, MacLean *et al.* categorized haptic design choices derived from literature and practice [11], but this is insufficient for actual practitioner use. The present analysis of design choices made by novice hapticians will inform future curricula, and enable development of better haptic and multisensory design tools.

### B. The Existing Haptic Design Tool Ecosystem

We see that hapticians have first co-opted mature non-haptic design tools, then dedicated haptic tools start to appear.

***Non-haptic designers have mature tools.*** The power and variety of design tools reflect the maturity of design theory in a field. Visual and web design are supported by numerous tools supporting different design activities, stages, and expertise levels. Tableau Public [31], D.tour [32], and Bri.colage [33] support ideation and *browsing* through galleries and resource/website recommendations. Balsamiq [34], Axure [35], and various domain-specific APIs (*e.g.,* D3.js [36]) support designers from *sketching* to *refining* and deployment, and enable *sharing* designs with team members and users. While many of these tools are oriented towards beginners, experts also commonly utilize their simple yet effective features and interface, particularly for early conceptual design.

***Non-haptic tools can be adopted for haptic design.*** Hardware platforms such as Phidgets (phidgets.com) [37], and more recently Arduino (arduino.cc) [38], the Maker movement and related rapid prototyping techniques have enabled and inspired fast iteration of haptic designs. Audio tools such as Audacity [39], SuperCollider [40], and PureData [41] are routinely used for vibrotactile synthesis [1], [42]. Many general-purpose programming languages such as C/C++ and Python as well as graphic and physics engines (*e.g.,* Unity [43]) are used for haptic and multisensory design [1].

While helpful, these tools do not adequately support hapticians. For example, interpolation algorithms designed for auditory signals fail in generating a perceptually meaningful *vibration* from vibrotactile sources [44].

Meanwhile, the practicalities of adapting the tools themselves to haptic media typically requires expert knowledge. For example, utilizing a generic physics engine to work with haptic hardware demands strong programming and physics skills.

***Haptic-specific tools are emerging from expert practice.*** Expert hapticians are developing tools to support their processes and needs. However, commercial tools and APIs such as Immersion Haptic Studio (vibrotactile effects) [15] and CHAI3D (force feedback) [16] have high learning curves. Our team has studied expert design practices for several years and developed online authoring tools for *browsing*, *sketching*, and *refining* vibrotactile stimuli [13], [42], [45].

Meanwhile, there are efforts to open the field to novices and designers with non-haptics backgrounds, via specialized hardware kits and graphical interfaces. With DIY (Do-It-Yourself) kits, individuals can make and modify haptic hardware from templates [46]–[49]. StereoHaptics adapts established audio-based tools to create, modify, and playback dynamic haptic media [50]. The TECHTILE toolkit allows recording and sharing of tactile material properties [51].

These forays into support of specific tasks and classes of haptic display are valuable. However, the haptic space, with its diversity of devices and points of bodily connection, combined with limited understanding of how to use it as a design medium, requires a more comprehensive mapping.

### C. Haptic Design for STEM Education

STEM education has long been a use case for haptics [8], [52]. The use of haptics as an adjunct learning modality is supported by complementary theories of embodied cognition, which suggest that multimodal and physical interactions assist in mentally simulating a concept, improving cognition [53], [54]. Researchers have developed haptic hardware and virtual learning environments for STEM learning [46], [52], [55], [56]. These have led to encouraging results for haptics in student learning and engagement (see [54] for a review).

Furthermore, STEM education provides a rich set of concepts to explore for haptic and multisensory design [57], established teaching practices and personal experiences to build upon, and potential for creative outcomes even with low-fidelity hardware [46]. These characteristics made STEM education an attractive theme for the WHC17 SIC and for our research goals.

| Device | Mechanism | Transmission | Workspace | Peak Force | Physics Simulation | Graphical Simulation | Communication Rate | Cost (USD) |
|---|---|---|---|---|---|---|---|---|
| **Hapkit** | 1-DoF Paddle | Capstan Drive | 50 mm arc | 4N | Arduino (C) | Processing (Java) | 160Hz | $100 |
| **Haply** | 2-DoF Pantograph | Geared | 70×140mm | 8N | Processing (Java) | Processing (Java) | 1000Hz | $150 |

Fig. 2. The SIC package supplied to the teams for the WHC17 SIC. (Upper left) contained Hapkit (middle) and Haply (right) hardware with their unique functionalities (bottom).

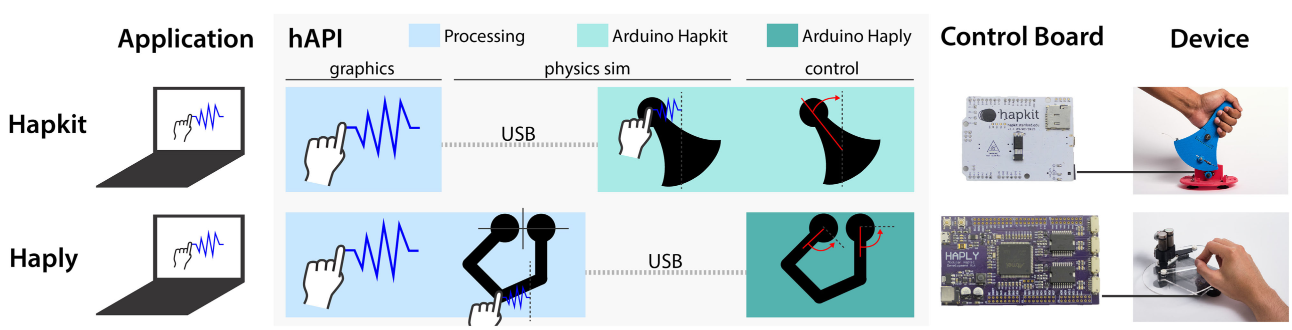



Fig. 3. Software architecture of the hAPI for both devices, Hapkit and Haply. Applications run on a host computer, *e.g.,* a laptop, while low-level software runs on Arduino control boards. The hAPI is distributed across an application API (Java/Processing, Box2D); a serial communication protocol; and a firmware implementation in C/Arduino for each device. Due to different serial rates, Haply physics simulations run in the application, while Hapkits run on the control board.

## III. A Design Package for Novice Hapticians

Novice hapticians need prototyping systems that are low-cost, open-source, easy to assemble and program, yet support some creative application design. We created the basic elements of a package that would allow us to study novice designers, with two existing low-cost devices and our own custom API.

Each package included (a) *physical components for two devices* (Fig. 2, shipped); (b) *software* (a modular haptics API for both devices, and example programs, provided online); and (c) *documentation* (shipped and online).

(a) and (b) are described below. For (c), our detailed documentation included hardware assembly instructions, software setup instructions, and a developer toolkit that provided details about the API architecture, an introduction to force-feedback haptic systems, and a set of example code to get the participants started. An archive of the support content can be found as originally delivered at *github.com/sic-whc2017*. The currently supported version can be found at *github.com/haplyhaptics*.

### A. Haptic Systems: Hapkit and Haply

***Hapkit:*** Hapkit's 1997 precursor [52] was updated in 2016 by Martinez *et al.* [46]. It has a 1-DoF (degrees of freedom) paddle mechanism, and a control board based on the Arduino Uno (8-bit, 16 MHz ATmega328) [38]. Designers communicate with the Atmega $\mu$controller from a host computer via USB and the Uno's UART FTDI USB-serial converter. The UART supported a rate of 160 Hz at the time of the SIC.

***Haply:*** The Haply system (Gallacher *et al.* [47]) is similarly open-source and customizable. It extends the basic kinematic structure of a 2-DoF pantograph [58] into a family of 1- to 3-DoF devices and kits [47], [59]. The Haply board is based on the Arduino Due (32-bit, 84 MHz Atmel SAM3X8E ARM); programming and communication use the Atmel's native USB interface. Host-Haply board communication is throttled in control firmware to 1000 Hz for communication stability.

***Architecture implications:*** Because of update rates and real-time needs (haptic rendering requires 500–1000 Hz, graphics 30–100 Hz), haptic-graphic display computation is often distributed across processors [60]. Fig. 3 compares the SIC device architectures. Each system's maximum inter-processor USB rate dictates where certain computations occur: host computer, *e.g.,* a laptop (*Processing-blue* in figure); or on a control board (*Arduino-green*).

Thus, Hapkit designers program graphics on the host but simulations on the control board, sending haptic state information upstream to the graphics module at 160 Hz. Haply designers can calculate physics on the host alongside the graphics module in a faster parallel process, then send rendering commands to the control board at 500–1000 Hz. The first development approach can simplify early code generation and communication, while

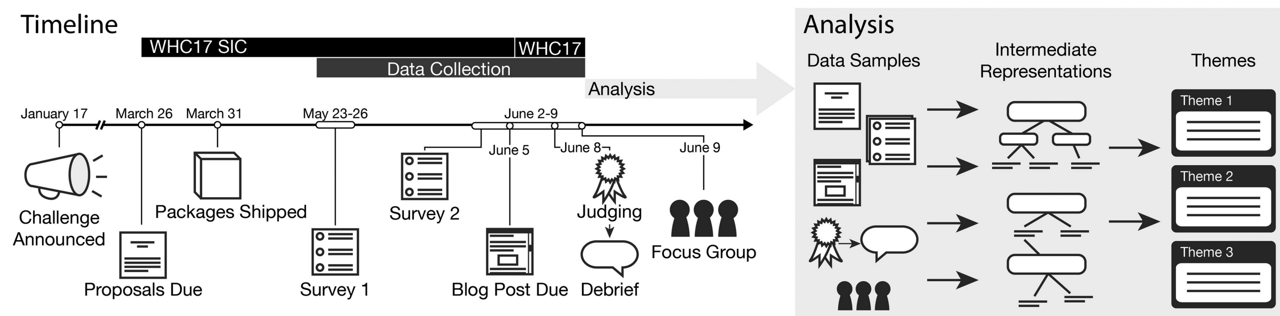



Fig. 4. Overview of challenge timeline, including data collection from various sources and our analysis process.

the latter can give greater control over model timing and is often used in production systems.

### B. hAPI: The Haptics Application Programming Interface

We needed a toolkit that simplified application development for force feedback devices like Hapkit and Haply, for novices to create engaging, albeit introductory, haptic environments.

***Approach:*** hAPI handles communication and commands between a host-computer-run application and the physical haptic device's controller board. It provides users with a modular API that 1) abstracts some low-level programming details (device setup, forward and backward kinematics), 2) is open-source to enable code customization, 3) usable across several platforms including Windows, macOS, Linux, and 4) modular for future integration with other force feedback devices.

hAPI differs from other tools, such as CHAI3D, by emphasizing user ability to adapt existing or develop entirely new vibrotactile or force-feedback designs. For example, the modular API allows configuration of sensors, actuators, and device mechanism with just a few lines of code.

***Development environment modules:*** hAPI[1] was developed in Java to access the Processing open-source development environment [61], accessing its support libraries and lowering the barrier to entry. hAPI uses the Processing Core library for communications and graphical development, and Box2D [62] for 2D rigid body simulation in 2D. Interaction forces are computed in Box2D, then output by hAPI to the device.

### C. Setting up the Development Environment: Hello Wall!

Teams first assembled physical devices from their kit (typically a ~1-hour job per device) and downloaded and installed a default hAPI configuration on host and controller. Teams tested their devices by installing and running supplied sample code, and consulting with the SIC organizers for support as needed. They were then ready to modify the default system to create their own setups and virtual environments.

## IV. WHC 2017 Student Innovation Challenge

The haptics conferences' challenge tradition has aimed to engage students with haptics technology and research, while bringing new ideas to the community. Our case study is based on the WHC17 SIC, a student-only submission track.

### A. Challenge Description and Support

Participants designed and prototyped an active learning environment for STEM topics using a low-cost haptic device. Definitions [67] and resources were linked from the challenge description website. Fig. 4 and Table I present the time course and details of the SIC and our data collection. The call for proposals and more lengthy challenge and study documents can be found in Supplement: Sections I,IV,V.

Following the SIC, announcement teams submitted proposals. Of 13 applicants, the SIC chairs selected 9 (Section IV-B) as finalists based on originality, feasibility, learning potential, accessibility, and adherence to submission guidelines. The number of teams was based on capacity and quality. Each finalist team received the SIC design package (Section III).

Teams were also given access to support resources, including: a Google Group for discussing issues with each other and the chairs, a template for their blog post, and the opportunity for financial support via travel grants awarded based on need. Chairs provided direct support via email for both technical issues and questions about the challenge for the SIC duration.

Finalists developed their applications over nine weeks, and supplied a final blog post explaining their concept and process. Finally, teams traveled to WHC17 and demonstrated their projects to the conference audience and a panel of expert judges to compete for awards (IV-C).

### B. Participant Teams

Our 9 teams (24 students) came from 7 countries (China, UK, Germany, Turkey, Mexico, USA, and Canada), each with 1-4 members. The SIC was open to secondary, post-secondary, or graduate students; teams all comprised either undergraduate or graduate students (9 undergraduate, 8 Master's, 5 Ph.D., 2 unknown). All had at least one member with engineering background. Some members came from other fields such as computer science or music technology.

Eight teams (22 students) completed the challenge by demonstrating their haptic environments at WHC17. We removed the ninth team (3 students) from our data analysis. The eight

[1] This library is renamed to haply-hAPI, to avoid a naming conflict with H3D HAPI engine that was identified after the challenge.

TABLE I
SUMMARY OF STUDY DATA SOURCES AND ANALYSIS METHODS. WE USED THEMATIC ANALYSIS TO FIRST ANALYZE EACH DATA SOURCE SEPARATELY, THEN INTEGRATE THOSE FINDINGS INTO COHERENT THEMES COVERING ALL DATA SOURCES [63], [64]

**PROPOSALS AND BLOG POSTS (per team)**

***Data Source:*** As part of the SIC deliverable, each team submitted an initial project proposal and a final online blog post detailing project learning goals, system design, hardware and software, to capture the haptic environment from their point of view.

***Analysis:*** We saved the proposal and blog posts in a pdf format and imported them into a qualitative data analysis software called QDA Miner Lite [65]. Two authors (SIC Analysis Team) separately read the proposals and blog posts, then created a list of codes (codebook) through discussion of variations observed in teams' design choices. The same two authors separately coded each blog post using the codebook, then discussed and consolidated these into a specification of the design choices for each team (Section VII-A).

**MID-DESIGN ONLINE SURVEYS (per student)**

***Data Source:*** We surveyed students 2 weeks before (midpoint) and immediately preceding WHC17 (final throes). Participation was voluntary, encouraged with €15 per survey. To minimize demand on participants, we focused our usability evaluation on factors of *ease of use* and *usefulness* [66]. Both surveys asked multiple choice questions about ease of use and usefulness of the SIC package and included open-ended questions about package shortcomings, challenges faced, and division of work among members. Survey 1 also asked about participant background and skills in software, hardware, and haptic development. Survey 2 inquired on diversion from initial vision and reasons for such changes. We received 7 and 11 responses to the two surveys respectively, with 6 participants completing both. See Supplement:Section V for exact questions.

***Analysis:*** Survey results were exported as a CSV file and then imported into QDA Miner Lite. The same two Analysis Team members aggregated survey ratings and multiple choice responses, and categorized and summarized textual responses. These categories were later adjusted, following analysis of data from the other sources.

**FOCUS GROUP (one session with all the teams)**

***Data Source:*** We conducted a 1-hour focus group at WHC17 after SIC winners were announced, with no compensation other than light refreshments. An SIC evaluation team member facilitated; the SIC chairs attended and participated in discussion. 7 student participants from 6 teams voluntarily attended the session, which was video recorded. See Supplement:Section IV for the focus group script.

***Analysis:*** Described next alongside judge interview analysis.

**SIC JUDGE INTERVIEWS (proxy and direct)**

***Data Source:*** To access expert evaluation of the team demonstrations, we had to avoid interfering with the SIC judges' discussion process and respect their time. A member of the SIC Eval Team silently shadowed 2 hours of judge meetings as they discussed the SIC projects in depth, then decided on awards. Recording this discussion was deemed inappropriate, but with permission, our team member narrated a secondary "gist" account to another team member in a proxy interview one day later. We also scheduled a 30-minute direct interview with one judge several weeks after the conference to triangulate insights. Both proxy and direct interviews were audio recorded.

***Analysis:*** All transcriptions were imported into QDA Miner Lite to be coded with the data from the other sources. Two Analysis Team members transcribed video and audio recordings from the focus group and judge interviews, then individually coded transcriptions. They applied descriptive keywords or phrases to one or multiple transcription sentences and paragraphs, reusing codes from blog posts and survey code books and creating new codes as needed [63], [64]. They discussed and consolidated codes, then coded the remaining transcriptions, additionally capturing patterns and observations in memos. With the full team, they linked codes through affinity diagrams and comparison.

**INTEGRATION OF ALL SOURCES**

After the initial open coding and memo writing phase, the full team identified the main directions for further analysis which is presented as the three themes in Section VII. Next, the same two Analysis Team members further used thematic analysis to code the data and link and aggregate the codes within these directions. They iteratively derived the content of the themes, checking them against the raw data at each step. The resulting quantitative data and code structures are presented as intermediate results in Section VI, and the three themes in VII.

teams and their projects are summarized in Table II, with longer descriptions and linked to their archival blog posts in Supplement:Section II.

### C. Conference Demonstration, Judges, and Awards

Student teams competed for three awards, two chosen by expert judges and one by audience vote (*People's Choice*). The best overall design, awarded by the judges, carried a cash prize of €1000. The runner-up and People's Choice awards each carried a cash prize of €750.

The SIC demos were presented publicly in a one-hour session. The SIC and another demonstration session were the full focus of the conference at that time, but teams were free to (and did) demonstrate outside of this period. Each demo had a booth with a 2'x4' table, a poster board and power supply.

Three judges, two expert in haptic technologies and one in education, were recruited to experience student demos, read student teams' blog posts, and then select an overall winner and runner-up. Judges first took notes, and then had a private discussion to choose the winners according to provided criteria: quality, effectiveness, and accessibility of the learning environment; effective use of haptics and device capabilities; and originality, creativity, and engagement.

### D. SIC Organization, Evaluation, and Analysis Teams

To avoid bias in data collection and analysis, for the period before SIC completion we divided our SIC researcher team (all of whom are co-authors) into two groups.

***SIC Chairs:*** Gallacher and Schneider prepared the SIC package, provided team support during their development stage, and organized SIC events at WHC17. They were assisted by co-chair Melisa Orta Martinez, and with high-level support by Challenge mentors MacLean and Allison Okamura. Orta Martinez and Okamura did not join the research team.

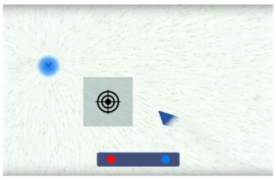
***Evaluators:*** MacLean and Seifi prepared and administered the surveys. MacLean recruited the SIC judges, silently attended their discussion session, and conducted the post-SIC focus group with SIC participants.

***Analysis (after SIC Awards):*** Seifi led data analysis, primarily assisted by Chun, an SIC participant, who contributed insider knowledge of his experience. All other authors also contributed. We deemed that the risk of bias was past, and Chun, as well as SIC chairs Gallacher and Schneider, had important insight into team experiences.

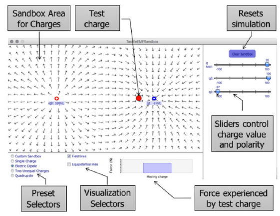
## V. METHODOLOGY

We captured teams' process and challenges during and immediately after their design work, with blog posts by each

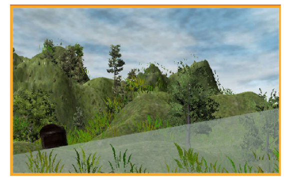
TABLE II
SUMMARY OF THE ENVIRONMENTS DEVELOPED BY THE EIGHT TEAMS THAT COMPLETED THE CHALLENGE (SUPPLEMENT:TABLE I)



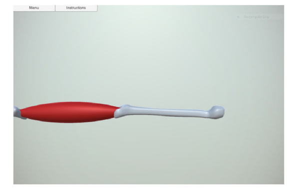
**(a)** $EMF - a_T$ (2 grads and 1 ugrad, Haply): Interactive environment where users could place electric charges in a 2D plane and feel the resulting electrostatic forces.



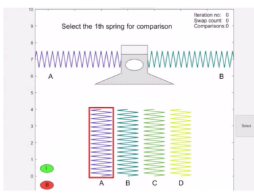

**(b)** $EMF - b_T$ (3 grads, Hapkit+Haply): Two sandbox environments. Users could place and configure electric charges in 1D and 2D, and feel electrostatic forces.



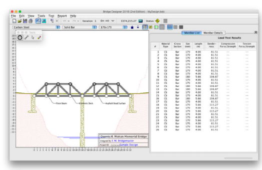
**(c)** $Movement_T$ (1 grad, 2 ugrads, Haply): Video game teaching buoyancy, wind, and friction forces. Users could transport a box and feel forces with varied settings.



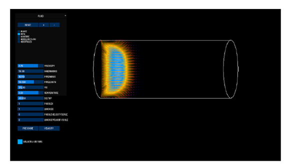
**(d)** $Muscle_T$ (3 grads, Hapkit): A series of lessons on muscle mechanics. Users could change the muscle length and excitation using Hapkit, then feel the resulting forces.



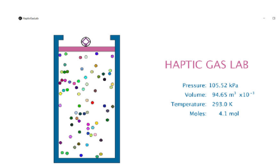

**(e)** $Sorting_T$ (1 grad, custom one-DoF haptic paddle): Sandbox environment to teach sorting algorithms. Users could feel stiffness of two springs, then sort them by stiffness values.



**(f)** $Bridge_T$ (1 ugrad and 1 industry practitioner, Haply): Bridge simulator. Users could design a bridge, move a truck across it, apply weights to find weak points, and feel normal force. The bridge would break for an over-capacity load.



**(g)** $Fluid_T$ (2 grads, Haply): Interactive environment for teaching Navier-Stoke equations for the motion of viscous fluid substances. Users could choose preset fluid environments and feel the fluid force at various points.




**(h)** $Gas_T$ (1 ugrad, Haply): Virtual cylinder for teaching ideal gas laws. Users could control position of a virtual piston at different pressures and temperatures, to feel contact forces from individual gas particles and overall piston gas pressure.

team, two surveys of individual members, and a post-challenge focus group. We accessed the SIC judges' discussion of the team demonstrations through a proxy mechanism, followed up with an interview with one judge. With thematic analysis, we analyzed data from each source and synthesized three coherent themes. Thematic Analysis is a qualitative method in which researchers iteratively apply open descriptive codes to the data, write memos and draw code diagrams to identify and refine patterned meaning in the data [63], [64].

Fig. 4 and Table I together summarize our data sources and details of our analysis procedures.

## VI. RESULTS I: USABILITY OF SIC PACKAGE

Due to the low number of responses, we present summary statistics of the usability ratings and do not apply statistical testing to the data (Fig. 5). We interpret these usability ratings based on open-ended survey question responses, the focus group, and interactions with SIC organizers.

***Haptic devices:*** Survey respondents rated Haply higher than Hapkit on both assessed metrics, *ease of use* and *usefulness*. Focus group comments attributed this difference to Haply's extra DoF (expanded interaction design possibilities) and software architecture (better support of computationally-intensive physical models). That is, the difference was a combination of a basic device attribute (versatility due to DoF), and a secondary, easily altered attribute (software architecture).

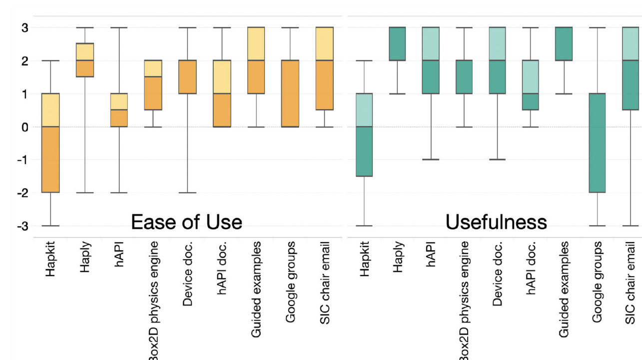



Fig. 5. Quantitative survey results: **ease of use** and **usefulness** ratings for SIC package components on a 7-point Likert scale (*-3:Not at all, +3:Very much so*). The plots show median, interquartile range, and max/min of 17 responses to Surveys 1 and 2 ([7,10] participants from [4,6] teams respectively). "*Doc.*" denotes *documentation*.

***hAPI and Box2D physics engine:*** Teams found these APIs effective, reasonably easy-to-use and adequately documented.

***Need for and usefulness of support:*** Overall, ratings suggest that device documentation, guided examples and SIC chair emails provided the most effective support of the SIC components (median rating $>=2$). Survey and focus group results indicated that these sources helped users set up the hardware (documentation), test its functionality (guided examples), and resolve their confusions with the hAPI and its documentation (SIC chair email). One team reported that access to the SIC chair for clarification and assistance was crucial.

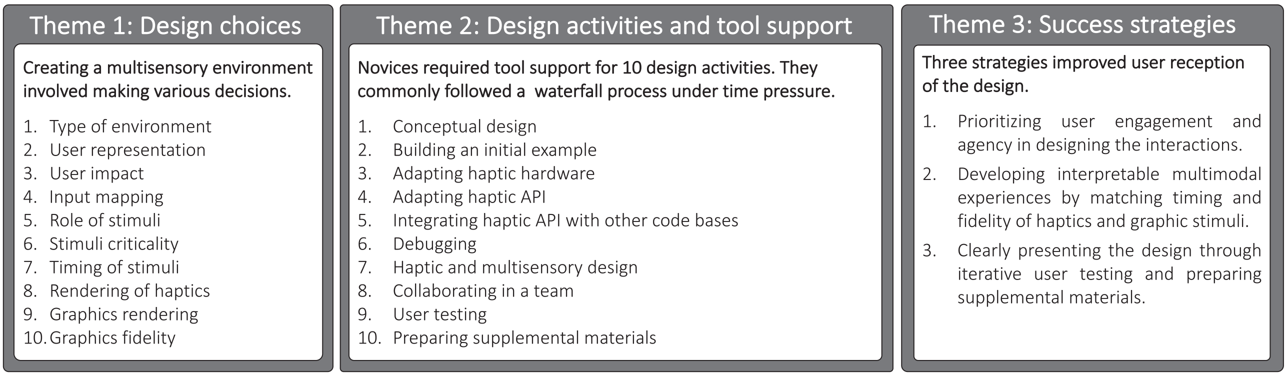



Fig. 6. We identified three main themes characterizing teams' 1) *choices*, 2) *activities*, and 3) *success strategies* in designing a multisensory environment. See Supplement:Fig. 1 for themes' full code hierarchy.

TABLE III
THEME 1 – SUMMARY OF TEAMS' DESIGN CHOICES, AND HARDWARE AND SOFTWARE THEY USED. THE TEN DECISION POINTS AND THEIR ALTERNATIVES, FOUND OR CONFIRMED IN OUR QUALITATIVE ANALYSIS, ARE DESCRIBED IN TABLE IV

| DESIGN CHOICES | *EMF-a*$_T$ | *Movement*$_T$ | *EMF-b*$_T$ | *Muscle*$_T$ | *Sorting*$_T$ | *Bridge*$_T$ | *Fluid*$_T$ | *Gas*$_T$ |
|---|---|---|---|---|---|---|---|---|
| **1. Environment type** | Sandbox | Game | Sandbox | Lesson | Hybrid | Sandbox | Sandbox | Lesson |
| **2. User representation** | Object | Object | Object | Themselves | Themselves | Object | Themselves | Themselves |
| **3. User impact** | Influence | Influence | Influence | Influence | Influence | Influence | Observe | Influence |
| **4. Input mapping** | Spatial→ Spatial | Spatial→ Spatial | Spatial→ Spatial | Spatial→ Spatial | Spatial→ Spatial | Spatial→ Non-Spatial | Spatial→ Spatial | Spat→ Spat; Spat→ None |
| **5. Role of stimuli** | Reinforce | Complement | Reinforce | Reinforce | Complement | Reinforce | Reinforce | Reinforce |
| **6. Stimulus criticality** | Secondary | — | Secondary | Primary | — | Same | Secondary | Secondary |
| **7. Timing of stimuli** | Synch | Synch | Synch | Synch | Synch | Synch | Synch | Synch |
| **8. Haptic Rendering** | Linear | Linear | Nonlinear | Linear | Preset | Linear | Nonlinear | Nonlinear |
| **9. Graphic rendering** | Dynamic | Dynamic | Static | Dynamic | Dynamic | Dynamic | Static | Dynamic |
| **10. Graphic fidelity** | Complex | Complex | Simple | Complex | Simple | Complex | Complex | Simple |
| **HARDWARE AND SOFTWARE CHOICES** | | | | | | | | |
| **Hardware** | Haply | Haply | Hapkit+Haply | Hapkit | Other [56] | Haply | Haply | Haply |
| **Graphics API** | Processing | Unity, uGUI | G4p, Grafica | Unity, X-Muscle, Blender | Matlab, Simulink | Bridge Designer 2016 | PixelFlow Processing Library | hAPI_Fisica |
| **Hardware Changes** | — | Added device stand | LCD & Arduino library | New handle & force sensor | — | — | — | — |

## VII. RESULTS II: THREE THEMES

Our data and analysis (Table I) produced the themes summarized in Fig. 6, which explicate (1) choices teams made (or did not make, or perceive as choices) in designing their environments; (2) context for these choices, *i.e.,* their teams' reported design activities; and (3) success strategies for haptic and multimodal design, based on judge discussions.

***Qualitative analysis code summary:*** These themes are derived from a total of 87 low-level codes to capture findings spanning all data sources, and organized hierarchically under the themes. These codes are diagrammed in Supplement: Fig. 1 to expose the detail underlying our thematic findings.

### A. Theme 1: The Learning Environments Varied along Interaction Paradigm as Well as the Role and Fidelity of Visual and Haptic Elements.

Through thematic coding and discussion of teams' design accounts as well as our observations, we identified ten choices as relevant and significant at a scope beyond that of individual projects (Table III). Each of these were also explicitly perceived as a decision point by at least one team.

Table IV elaborates on these choices. The first four are related to interaction design in any modality; the rest are specific to multimodal integration. These choices were not described or taught to the SIC teams or judges, but judge comments suggested that they impacted demo effectiveness.

### B. Theme 2: Designing a Multisensory Environment Involved Many Activities, Some of Which the Teams Skipped or Rarely Iterated on.

Ten[2] activities and associated challenges emerged from teams' descriptions of their design process in the focus group and survey data (Fig. 7).

***1. Conceptual design:*** All teams proposed some interactions for their environment at the proposal stage, but they

[2] The finding of 10 items under both Themes 1 and 2 is coincidental.

TABLE IV

THEME 1 – ELABORATION OF THE 10 DESIGN CHOICES IDENTIFIED. 1-4 AND 8-10 EXTEND THOSE IDENTIFIED IN [11]

**1. Type of environment** [*lessons, sandbox, game, hybrid*]
$Muscle_T$ developed four preset lesson environments, each centered around a question (*e.g.,* $Muscle_T$ *[Blog post]: "How do muscle forces change as a muscle fatigues?"*). Five teams adopted a sandbox approach where users could customize various environment elements (*e.g.,* number and magnitude of electrical charges). $Movement_T$ incorporated game mechanics, such as a storyline and rewards for user engagement; $Sorting_T$ used a hybrid of a sandbox and lessons.

**2. User representation in the environment** [*themselves, an object*]
Half of the environments portrayed users as themselves within the virtual environments (*e.g.,* the user felt forces from a virtual spring applied to their hand). The other half portrayed the user as an object in the environment (*e.g.,* a test charge "felt" forces from an electric field).

**3. User impact on the environment** [*influence, observe*]
In 7 out of 8 environments, user input could influence the environment's state (*e.g.,* pushing on a spring made it compress). In the exception, $Fluid_T$, the user could feel environment forces without impacting fluid flow (observe), a configuration possible only in a virtual environment.

**4. Input mapping** [*spatial→spatial, spatial→nonspatial, spatial→none*]
Six teams mapped device spatial input parameters to spatial haptic display parameters. Exceptions were $Bridge_T$, which mapped Haply's second input to a non-spatial parameter (truck weight); and $Gas_T$, which did not utilize Haply's second input.

**5. Roles of multimodal stimuli** [*reinforcing, complementary*]
In six projects, haptic and visual stimuli presented information that *reinforced* [11] the same percept (*e.g.,* $EmF - a_T$ displayed electromagnetic field strength at a specific point via *both* Haply's output force and the density of graphical lines). In $Sorting_T$ and $Movement_T$, visual and haptic modalities *complemented* one another (*e.g.,* spring compression was displayed graphically, spring force haptically).

**6. Stimulus criticality** [*primary, secondary*]
When haptic sensations reinforce another modality, they can be the primary or secondary source of information for a percept [11]. In four environments, the visual modality was *primary*: *e.g.,* graphics showed fluid forces in the whole $Fluid_T$ environment at once, while the *secondary* haptic modality rendered force at a single point location through Haply). For $Bridge_T$, the two modalities played a similar role; for $Muscle_T$, haptics was primary. The other two teams presented complementary information in the visual and haptic modalities.

**7. Stimuli timing** [*initial, followup*]
We did not find any variation or indication of experimentation in temporal sequencing of multimodal stimuli [11]: related visual and haptic stimuli were always synchronized in time.

**8. Rendering of haptics** [*pre-designed, model-based, data-driven*]
All teams used a mathematical model to render haptic stimuli, but tweaked models to different extents to make them perceptually distinct. At one extreme, $Sorting_T$ used preset model parameters (*e.g.,* spring constants). Four teams linearly scaled model output to fit device force output. $Gas_T$, $Fluid_T$, and $EMF - b_T$ enhanced perceptibility via nonlinear parameters and output adjustments. No team used preset sensations or rendered output based on data from real-world interactions.

**9. Graphic rendering** [*static, dynamic*]
Teams adopted either a static graphical information representation which did not change during an interaction (2 teams, *e.g.,* $EMF - b_T$); or a dynamic one that updated based on user input (6, *e.g.,* $Gas_T$).

**10. Graphic fidelity** [*simple, complex*]
Five teams developed high fidelity graphics to accompany haptic information (Table II), while the other three purposefully lowered graphic fidelity to a simple, cartoonish representation for consistency with the haptic sensation fidelity (Table II).

rarely iterated later on their initial ideas. We see this in Fig. 7, created by comparing proposals and blog posts: for most teams, conceptual design (light blue) happens only in early stages. Only $EMF_{aT}$, $Movement_T$ and $Sorting_T$ revised environment mechanics and user interactions. This lack of revision is notable: at proposal stage, the information teams had about the devices was limited to what was available on the device websites. They named DoF as the main criteria for their choice of device and interactions.

Teams did not report using any learning resources or online repositories for inspiration beyond technical materials required for their development (*e.g.,* API documentation, Stanford online haptics course [68], STEM books). They made these design choices based on personal intuition and experience, apparently without explicitly considering alternatives.

***2. Build an initial example:*** All teams could assemble the hardware and run a set of "hello-wall" examples provided in the SIC packages (dark blue in Fig. 7). High ratings for the assembly documentation and example codes reported earlier suggest their effectiveness for helping teams in device setup. However, even with these resources, verifying correct software setup and device output was sometimes challenging.

> *P9 [focus group]: "The biggest challenge was the Hello World. We kept thinking why it's not working. You can see the device is detected, but where's the feedback? Is it supposed to be like this?"*

Lack of a representation or descriptive language to document and transfer a device's "feel" contributed to this challenge.

***3. Adapting the hardware:*** Four teams customized the SIC haptic devices to: 1) include a custom 3D-printed handle and pressure sensor ($Muscle_T$), 2) add an Arduino LCD screen for text display ($EMF - b_T$), 3) adjust device size ($fluid_T$), and 4) add a stand for Haply on a tablet ($Movement_T$). The open-source nature of the devices facilitated these modifications. No challenge was reported for this activity.

***4. Adapting hAPI:*** All teams utilized the provided device API (hAPI); two adapted it. In the surveys, teams rated the device setup, serial port communication, and forward and backward kinematics functionalists of hAPI as especially effective. hAPI supported adaptations: $EMF - b_T$ wrote additional scripts to obtain more precise sensor readings and damp undesirable device vibrations in application edge cases. $Movement_T$ rewrote the Java-based hAPI routines in C# to use it with Unity.

***5. Integrating hAPI with other code bases:*** To develop a multimodal environment, all teams needed to integrate hAPI with a physics and graphics engine. The SIC package facilitated this with a customized physics engine; but, due to its late release (two weeks into the challenge) plus some teams' prior experience with other languages, only one utilized it ($Gas_T$).

Integration with a 3rd-party physics engine involved notable effort. $Movement_T$ rewrote the hAPI in C# to integrate it with the Unity physics engine. $EMF - b_T$ wrote their own scripts for their physics and graphics components. $EMF - a_T$

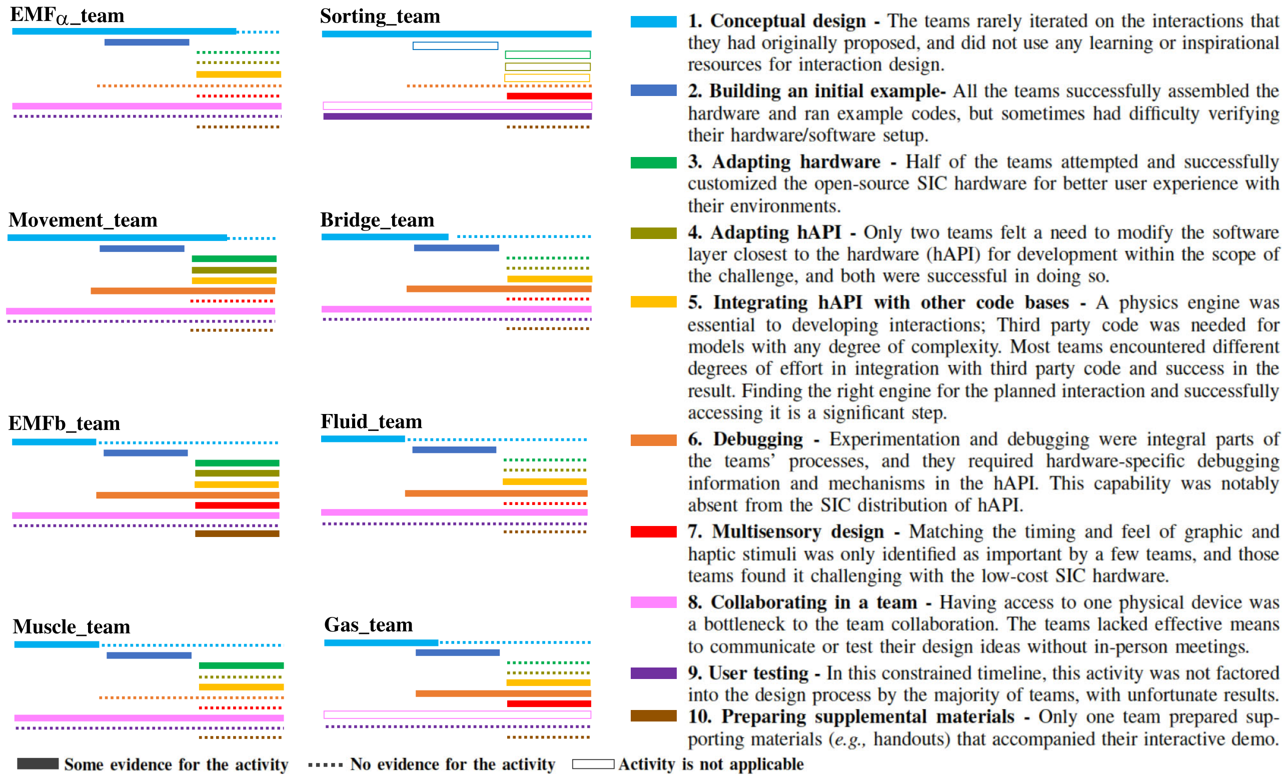



Fig. 7. Design activities reported by SIC teams. Line location and length indicate timing and duration of each design activity which we estimated based on our data. $Sorting_T$ and $Gas_T$ have hollow boxes for activities inapplicable to their cases. *i.e.,* $Sorting_T$ used their own custom hardware and API and both teams were composed of one member (no collaboration).

integrated hAPI with a third-party physics engine through trial and error, and $Fluid_T$ contacted the SIC chair for help in a similar API integration case. At least four teams found this activity challenging; all ultimately succeeded. The open-source nature of hAPI and our technical support facilitated integration. Sparsity of hAPI examples and documentation on 3rd-party engine integration was a bottleneck.

***6. Debugging:*** Five teams reported substantial trial-and-error as they learned to relate command input to haptic output (prominent orange line in Fig. 7), during which their attempts sometimes led to device overtaxing and failures. Three teams requested a set of hardware-specific debugging mechanisms: 1) a detailed specification of the device input range, 2) a "safe mode" environment that warns for undesirable input, 3) a preview of the device status for given input values, and 4) meaningful error messages for faulty hardware or software configuration (the distribution package provided Processing error messages which were not sufficiently informative).

***7. Haptic and multisensory design:*** Matching the timing and feel of visual and haptic effects were the core of this activity. The SIC architecture and API were optimized to minimize processing delays, yet the external mathematical and graphics APIs (not designed to support haptics) sometimes became a processing bottleneck. $Fluid_T$ had problems incorporating a computationally-intensive fluid model; $Movement_T$ faced low refresh rates using Unity for complex graphic scenes. A small delay in haptic and graphic timing led to *"unintepretable haptic noise"- [proxy interview].*

Matching the "feel" of the visual and haptic effects received less team attention than temporal simultaneity (dashed line in Fig. 7), but was tough for those who tackled it. The main difficulty was that the SIC hardware had too little force range to reflect a physical phenomenon (*e.g.,* gas interactions). Three teams found different solutions. $Sorting_T$ simply programmed two force profiles, but acknowledged its limit in generalization. $EMF - b_T$ used filtering and damping techniques. $Gas_T$ exploited Haply's fast processing rate (*i.e.,* high temporal resolution) to display hundreds of haptic bursts, simulating contact forces from gas particles on a cylinder wall. $Gas_T$ also utilized a simple cartoonish representation to match the haptic fidelity.

> *P20 [focus group]: "I had a lot of iterations ... I slowed particles down, I made them bigger and heavier ... I would have to kind of tune the haptics side every time because it didn't necessarily produce the same kind of intuitive haptic response that you were seeing on the screen. So matching them [graphics and haptics] up ... forced a lot of tweaking."*

***8. Collaborating in a team:*** Teams rarely commented on the social factors that impacted their design. In an exception, three teams noted that having one device was a "bottleneck" in their development, requiring them to frequently meet in-person.

$Fluid_T$'s two members had to work from different continents, one assembling hardware and the other programming. Iterations were difficult, limited to verbal description of the

TABLE V
THEME 3 SUMMARY – THREE SUCCESS STRATEGIES DERIVED FROM THE JUDGES' EVALUATION OF THE SIC DEMONSTRATIONS

| |
|---|
| **Strategy 1: Support user engagement and agency** in a haptic virtual environment by exaggerating, complementing, or simplifying real world interactions and sensations. |
| **Strategy 2:** For interpretable interactions on low-cost haptic hardware, **use cartoonish yet dynamic graphics** that can match the haptic fidelity and tune output of mathematical models for enhanced haptic perception. Use the time dimension effectively; design for individual as well as aggregate temporal events and match the timing of haptic and visual stimuli. |
| **Strategy 3: Understand your domain.** Learn about the existing ecosystem of physical and digital tools in your application domain and use those as supplemental materials to situate your design for the users. Seek and incorporate user feedback throughout your design process, not only at the end. |

output and its problems. While remote collaboration is challenging for any hardware project, an added challenge in haptics is that the output cannot be captured and communicated to collaborators.

> *P4 [Survey 2]: "Whenever my partner would suggest software changes, instead of knowing the results of the change immediately and tangibly, she'd have to discover the results minutes or hours later via a text description."*

***9. User testing:*** Only $Sorting_T$ tested their design with users. The other teams reported user testing as a future step and skipped it due to time constraints.

***10. Preparing supplemental materials:*** Only $EMF - b_T$ designed posters and handouts to reinforce lesson concepts and guide users through the interface and interactions.

Next, we explain how the teams' design choices and activities reflected in the judges' evaluation of their environments.

### C. *Theme 3: Judges Valued Interactions That Complement Everyday Experiences, are Interpretable, and are Clearly Communicated in the Context of Existing Real-World Ecosystems.*

We present the criteria that emerged from our analysis of the judges' discussion below, and summarize the strategies that the judges deemed as successful in meeting these criteria in Table V.

***1. Useful and engaging interactions:*** Effective virtual environments enabled physical interactions beyond what was possible in the real world, and supported user agency and engagement. The education judge noted that physical laboratory demonstrations have limited ability to cover all science concepts, and emphasized the need for experiences that complement rather than replicate existing real world solutions.

Our analysis of judges' observations suggests at least three ways in which the SIC demos achieved novel user experiences and engagement with their virtual environments: by 1) exaggerating physical effects so as to be perceivable (*e.g.,* slowing down gas interactions), 2) enabling users to *observe* forces without *influencing* the environment dynamics (*e.g.,* feeling fluid pressure with a virtual probe), 3) providing a game or sandbox environment in which users could reconfigure parameters of a complex mathematical model to gradually build intuition (*e.g.,* electromagnetic fields and forces).

***2. Interpretable multimodal experiences:*** The judges often discussed whether a haptic rendering accurately reflected a physical phenomenon and was in harmony (matched) with the visual stimuli. They noted that simply implementing a mathematical formula and scaling the output value to the device output range was not enough to reflect the richness of the model, that a match in timing of visual and haptic stimuli was critical, and high-fidelity "*mesmerizing*" visuals could overshadow low-fidelity haptic sensations and hinder their interpretability.

> *J1 [direct interview]: "So the graphics were gorgeous. The haptics didn't match the graphics part of it at all. It just felt bumpy and like a lot of resistance and hard to tell what it is you were feeling... and it was not clear what are we supposed to be learning about other than just being mesmerized by a beautiful display."*

They praised simple representations, nonlinear force scaling with filtering and damping scripts, and use of aggregate/composite temporal events to deliver rich haptic sensations to the users (see Section VII-B for more details).

***3. Clear design presentation:*** Judges agreed on the need for clearer articulation of and framing of lessons around learning goals and interactions. In some case, teams were unaware of the learning value of their environments and/or could not adequately justify their design choices.

The more successful presenters leveraged their knowledge of the education ecosystem. The winning team utilized prior experience as teaching assistants to develop their lesson plan. Further, they designed supplemental materials (posters and sample handouts) that *"maybe subliminally helped us get all prepared," J1 [direct interview].*

Sharing environments with people outside the design team also improved design and delivery of a demo. While all teams reported an intention to test their environments with users in their blog posts, they planned it as a last step for their design and most did not achieve it in their timeframe. Only one team was able to test their environment with potential users, reporting revision of their conceptual design and haptic rendering based on user feedback. This team won the SIC People's Choice award, possibly not coincidentally.

## VIII. DISCUSSION

### A. *Current Novice Haptician Practices Relative to Experts*

We review the novice haptician design processes observed in our study ("as-is") and compare these practices to those of expert hapticians to highlight existing gaps.

***Basics: All teams could configure and adapt the SIC package, and build a working multimodal environment.*** We attribute this to the transparent and modular design of the hAPI and the SIC hardware, which enabled teams to integrate external hardware and graphics APIs with the SIC package and build at least a basic working environment with multiple display modalities. This is a non-trivial achievement given their limited background as hapticians, lack of prior experience with the package, and short timeline.

***Obstacles: Like experts, these teams struggled to synchronize haptics with visual stimuli, and to debug and communicate designs with teammates.*** Aligning cross-modal stimuli required integration of hardware with computational models and external APIs. Inability to preview content hindered rapid, safe technical content iteration, while lack of simulation capabilities deeply complicated team communications, particularly given a single device. These challenges are not specific to novices. The vertical nature of haptic development and dependencies throughout the hardware and software pipeline plagues experts' prototyping, debugging, and sharing [1].

***Process: In contrast to experts, novices skipped or skimped on design activities which experts engage in.*** These included pre-design browsing, significant iteration during conceptual and multimodal stimuli design, and getting feedback [27]. Our novices relied largely on personal experience to devise interactions, without reference to the design examples (even if in other modalities) which experts use for conceptual inspiration. They did "sketch," but this was largely for technical development, not conceptual design or to address feedback. While all teams did strive for temporal visuohaptic synchronization, five teams attempted no other aspects of perceptual fusion. In contrast, experts commonly focus on perceptual fusion of the senses as a whole [69] and seek perceptual rules such as substitution and reinforcement to mitigate haptic output constraints [11]. Finally, 7 of 8 teams did not report seeking external input before their SIC presentations, where we observed difficulties in conveying their vision and design choices to the judges and audience – a mistake with serious consequences for professionals.

These results suggest that hAPI's principles do lead to customizable hardware and API with a short learning curve. Yet, both novices and experts need further support for rapid prototyping, debugging, and remote sharing. In addition, novices would benefit from a roadmap to an iterative design process, with specific support for conceptual and multisensory design as well as sharing and delivery to users and target audience.

### B. What-If: Envisioned Haptic Design Resources and Process

Effectively supporting novices and experts requires far more than implementing software features in existing APIs: we need a comprehensive agenda for theory, infrastructure, and tool development. Below, we examine each, then revisit them in our opening scenario.

***Need 1. Theory and practical guidelines for haptic design***

*Haptician activities and process:* This study and previous work on expert hapticians are the groundwork for a prescriptive theory of haptic design. We envision a practical design roadmap, addressing questions such as: What are similarities and differences between designing for haptics and other modalities? How are design activities best sequenced and linked? What are best practices in modality fusion?

*Haptician design choices:* This research contributes a characterization of the types of questions and choices a haptician needs to address, informed by emerging haptic design theory, related formalizations in other fields, and our present findings of the inaccessibility of these choices for novices. Future studies need to build on these parameters (Table IV) by studying haptic design in other contexts.

***Need 2. Infrastructure and content for design tools***

*Curate design examples and content galleries:* Our community needs to further invest in developing repositories as a shared knowledge resource. Needs range from haptic effects compiled by a single research team [42], [70] to community-sourced [10] or domain-specific repositories ([57] for education). Their development needs to respect some emergent requirements from this and previous studies [1], [10], [27], [42], such as: (a) ideation in the early design stages demands efficient browsing; (b) example reuse and customization requires access to compatible source files and editing tools; (c) mining a collection for design patterns requires effective search, filtering and other types of selective access.

*Enable sharing of haptic experiences by developing representations:* Haptic hardware is diverse and often custom-built. As a community, we need means to record, transfer, and evaluate haptic experiences without access to identical hardware. Developing representations and proxies in visual, auditory, and haptic modalities as well as haptic vocabulary for efficient communication are promising directions [45].

*Establish an active haptic design ecosystem:* By necessity, haptic design and engineering is usually done in very local contexts, without access to the experience of the larger community. Progress would be accelerated by online information hubs for novices, online and offline haptic design workshops and classes, and discussion forums (like Stack Overflow [71]) for haptic interaction and engineering design.

***Need 3. An ecosystem of haptic authoring tools***

To support novices, authoring tools should utilize developments in haptic theory and infrastructure.

*Support different design stages, activities, and user groups:* Haptic design needs a range of tools for different activities. *Browsing* and *conceptual design* can be best supported through design galleries with efficient browsing and search functionality as well as learning resources on the conceptual design choices. *Sketching* and *refinement* tools need to provide real-time feedback, enable direct manipulation, and incorporate templates and defaults based on best practices and examples. While iterating on a design, hapticians need debugging tools that "preview" output, have a "safe mode" preventing dangerous device configurations, and visualize latencies along the hardware and computational pipeline. Tools for all design stages should enable prototype sharing, but may require different mediums and representations depending on design activity and audience. For example, for debugging and diagnosis, video galleries may be effective whereas end-user testing may benefit most from other visual and auditory proxies. While these tools can support experts, novice tools should provide more design tips and templates and hide advanced features from accidental misuse but provide a straightforward path to the full interface as novices mature to become experts.

*Embed design space & process information in tools:* Tools can reinforce an efficient design process in addition to supporting one or more specific design activities. Explicitly, their tutorials and learning resources can showcase best practices and effective workflows with the tool and its role within a larger design process. Implicitly, the tools can incorporate design parameters as templates and defaults, and enable progression through various design activities, *e.g.,* by supporting import and export of design assets/content across design stages.

***Scenario: How would Alex and his team benefit from the above haptic theory, infrastructure, and tools?*** *For an engineering course project, Alex and Jing decide to develop a haptic game environment for STEM education. Through haptic community blogs, postings, tutorials and repositories, they learn about tools suitable for their intended technology (serial 2D force feedback), and a workflow that shows how to use the tools. Following a recommended design roadmap, they begin with the conceptual design and sketching tools marked as best for novices. They explore design examples for similar haptic devices an/or application domains and sketch several low-fidelity prototypes, which vary in conceptual and multisensory design choices.*

*As they refine some of these sketches and move to high-fidelity versions, they continuously experiment with input and output values, check latencies in the debugging console, and adjust multimodal synchronization. They preview output values in other proxy modalities and share them with team members, the haptic expert community, or end-users for debugging and feedback. They learn about advanced tool features through community postings, learning resources and personal explorations, and adapt their tool configurations for their own purposes. During the end-term demo, they could explain their design choices, and critique them in light of user feedback.*

## IX. Conclusions

In this paper, we investigated novice hapticians' design practices, and their tool and theory needs as a primary step toward supporting multisensory design by thousands of practitioners (*e.g.,* interaction designers, domain technologists) who are interested in haptics. We presented an open-source package for novices composed of haptic hardware, API, and documentation and reported on a 9-week study of nine student teams who used the package to design a multisensory environment for STEM education during the IEEE WorldHaptics Conference 2017 Student Innovation Challenge. Our analysis of quantitative and qualitative data from the student teams and SIC judges yielded three main findings: (a) 10 parameters characterizing the teams' interaction design choices and variations in creating their environments, (b) 10 activities that are important for haptic design, yet were often skipped by these novices, and (c) three strategies that led to successful design and delivery of the multisensory environments to the SIC judges and audience.

Comparing these novice teams' practices to expert processes from previous studies, we have highlighted effective aspects of our open-source package and outlined further needs for supporting novices and, potentially, experts. Novices especially suffered from a missing characterization of interaction design choices and process. We feel that this information should be embedded in novice design tools and delivered in haptic and multisensory design workshops and tutorials. Both novices and experts need substantial infrastructure, tool, and process support for browsing of existing designs, rapid prototyping, debugging, and sharing of their multimedia content.

With this paper, we call on the haptics community to build on emerging haptic design theory, infrastructure, and tools to support hapticians in a wide range of contexts.

## Acknowledgment

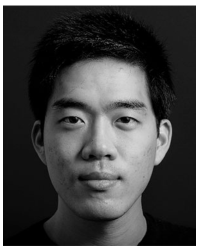
The authors would like to thank Dr. Allison Okamura and Melisa Orta Martinez for assistance with the Student Innovation Challenge, the SIC teams and judges for sharing their experiences, the World Haptics'17 organizers for their support, and NSERC Canada for funding this research.

## References

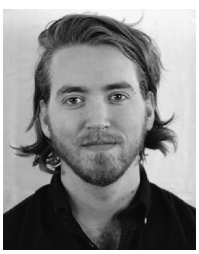
[1] O. Schneider, K. E. MacLean, C. Swindells, and K. Booth, "Haptic experience design: What hapticians do and where they need help," *Int'l J. Human-Comput. Stud.*, vol. 107, pp. 5–21, 2017.
[2] F. G. Hamza-Lup and I. A. Stanescu, "The haptic paradigm in education: Challenges and case studies," *Internet Higher Edu.*, vol. 13, no. 1-2, pp. 78–81, Jan. 2010.
[3] R. L. Davis, M. O. Martinez, O. Schneider, K. E. MacLean, A. M. Okamura, and P. Blikstein, "The haptic bridge: Towards a theory for haptic- supported learning," in *Proc. Conf. Interact. Des. Children.*, 2017, pp. 51–60.
[4] W. McMahan *et al.*, "Tool contact acceleration feedback for telerobotic surgery," *IEEE Trans. Haptics*, vol. 4, no. 3, pp. 210–220, Jul.–Sep. 2011.
[5] O. A. Van der Meijden and M. P. Schijven, "The value of haptic feedback in conventional and robot-assisted minimal invasive surgery and virtual reality training: a current review," *Surgical Endoscopy*, vol. 23, no. 6, pp. 1180–1190, 2009.
[6] D. Wang *et al.*, "Idental: A haptic-based dental simulator and its preliminary user evaluation," *IEEE Trans. Haptics*, vol. 5, no. 4, pp. 332–343, Fourth Quarter 2011.
[7] I. Choi, E. Ofek, H. Benko, M. Sinclair, and C. Holz, "Claw: A multifunctional handheld haptic controller for grasping, touching, and triggering in virtual reality," in *Proc. CHI '18*, 2018, pp. 654:1–654:13.
[8] K. E. MacLean and V. Hayward, "Do it yourself haptics: Part II [Tutorial]," *IEEE Robot. Autom. Mag.*, vol. 15, no. 1, pp. 104–119, Mar. 2008.
[9] H. Seifi, C. Anthonypillai, and K. E. MacLean, "End-user customization of affective tactile messages: A qualitative examination of tool parameters," in *Proc. IEEE Haptics Symp.*, 2014, pp. 251–256.
[10] H. Seifi *et al.*, "Haptipedia: Accelerating haptic device discovery to support interaction & engineering design," in *Proc. CHI '19*. 2019, pp. 558:1–558:12.
[11] K. E. MacLean, O. Schneider, and H. Seifi, "Multisensory haptic interactions: Understanding the sense and designing for it," in *Proc. Handbook Multimodal-Multisensor Interfaces*. ACM Books, 2017, pp. 97–142.
[12] L. Jaebong, R. Jonghyun, and S. Choi, "Vibrotactile Score: A score metaphor for designing vibrotactile patterns," in *Proc. World Haptics'09*, 2009, pp. 302–307.
[13] O. S. Schneider, A. Israr, and K. E. MacLean, "Tactile animation by direct manipulation of grid displays," in *Proc. ACM Symp. User Interface Softw. Technol.*, 2015, pp. 21–30.
[14] H. Seifi, M. Chun, and K. E. Maclean, "Toward affective handles for tuning vibrations," *ACM Trans. Appl. Perception*, vol. 15, no. 3, pp. 22:1–22:23, Jul. 2018.
[15] Haptic experience. Accessed: Nov. 9, 2018. [Online]. Available: https://www.immersion.com/haptic-experience/
[16] CHAI3D. Accessed: Nov. 9, 2018. [Online]. Available: http://www.chai3d.org/

[17] M. Resnick, B. Myers, and K. e. a. Nakakoji, “Design principles for tools to support creative thinking,” in *Proc. NSF Workshop Rep. Creativity Support Tools*, 2008, pp. 25–36.
[18] B. Buxton, *Sketching User Experiences: Getting the Design Right and the Right Design*. San Mateo, CA, USA: Morgan Kaufmann Publishers Inc., 2007.
[19] H. Sharp, J. Preece, and Y. Rogers, *Interaction Design: Beyond Human-Computer Interaction*. Chichester, UK: John Wiley & Sons Ltd, 2019.
[20] C. L. Dym, A. M. Agogino, O. Eris, D. D. Frey, and L. J. Leifer, “Engineering design thinking, teaching, and learning,” *J. Eng. Edu.*, vol. 94, no. 1, pp. 103–120, 2005.
[21] F. Danieau, J. Fleureau, P. Guillotel, N. Mollet, M. Christie, and A. Lecuyer, “Toward haptic cinematography: Enhancing movie experiences with camera-based haptic effects,” *IEEE MultiMedia*, vol. 21, no. 2, pp. 11–21, Apr.–Jun. 2014.
[22] P. Guillotel, F. Danieau, J. Fleureau, and I. Rouxel, “Introducing basic principles of haptic cinematography editing,” in *Proc. Eurographics Workshop Intell. Cinematography Editing*, 2016, pp. 1–7.
[23] E. Gunther, G. Davenport, and S. O’Modhrain, “Cutaneous grooves: Composing for the sense of touch,” in *Proc. New Interfaces Musical Expression*, 2002, pp. 73–79.
[24] C. Moussette, S. Kuenen, and A. Israr, “Designing haptics,” in *Proc. Text Encoding Initiative*. New York, New York, USA: ACM Press, Feb. 2012, p. 351.
[25] J. Corbin and A. Strauss, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, 3rd ed. New York, NY, USA: Sage Publications, Inc., 2008.
[26] O. S. Schneider and K. E. MacLean, “Improvising design with a haptic instrument,” in *Proc. IEEE Haptic Symp.*, 2014, pp. 327–332.
[27] O. S. Schneider and K. E. MacLean, “Studying design process and example use with Macaron, a web-based vibrotactile effect editor,” in *Proc. IEEE Haptic Symp.*, 2016, pp. 52–58.
[28] T. Munzner, *Visualization Analysis and Design*. United States: AK Peters, 2014.
[29] D. Norman, *The Design of Everyday Things: Revised and Expanded Edition*. New York, NY, USA: Basic Books (AZ), 2013.
[30] L. Jones, “News from the field: Courses in haptics,” *IEEE Trans. Haptics*, vol. 7, no. 4, pp. 413–414, Oct.–Dec. 2014.
[31] Free data visualization software | tableau public. Accessed: Nov. 8, 2018. [Online]. Available: https://public.tableau.com/en-us/s/
[32] DTour TV. Accessed: Nov. 8, 2018. [Online]. Available: https://www.dtourtv.com/
[33] R. Kumar, J. O. Talton, S. Ahmad, and S. R. Klemmer, “Bricolage: Example-based retargeting for web design,” in *CHI ’11*. New York, NY, USA: ACM, 2011, pp. 2197–2206.
[34] Balsamiq. rapid, effective and fun wireframing software.—balsamiq. Accessed: Nov. 7, 2018. [Online]. Available: https://balsamiq.com/
[35] Prototypes, specifications, and diagrams in one tool—axure software. Accessed: Nov. 8, 2018. [Online]. Available: https://www.axure.com/
[36] M. Bostock, V. Ogievetsky, and J. Heer, “D3: Data-driven documents,” *IEEE Trans. Visualization Comp. Graph. (Proc. InfoVis)*, vol. 17, no. 12, pp. 2301–2309, Dec. 2011.
[37] S. Greenberg and C. Fitchett, “Phidgets: Easy development of physical interfaces through physical widgets,” in *Proc. ACM Symp. User Interface Softw. Technol.*, 2001, pp. 209–218.
[38] Arduino - home. Accessed: Nov. 7, 2018. [Online]. Available: https://www.arduino.cc/
[39] Audacity®— free, open source, cross-platform audio software for multi-track recording and editing. Accessed: Nov. 8, 2018. [Online]. Available: https://www.audacityteam.org/
[40] Supercollider. Accessed: Aug. 1, 2018. [Online]. Available: https://supercollider.github.io/
[41] “Pure data — Pd community site.” 2018. [Online]. Available: https://puredata.info/
[42] H. Seifi, K. Zhang, and K. E. MacLean, “VibViz: Organizing, visualizing and navigating vibration libraries,” in *Proc. World Haptics*, 2015, pp. 254–259.
[43] Unity. Accessed: Nov. 9, 2018. [Online]. Available: https://unity3d.com/
[44] B. Clark, O. S. Schneider, K. E. MacLean, and H. Z. Tan, “Predictable and distinguishable morphing of vibrotactile rhythm,” in *Proc. World Haptics*, 2017, pp. 84–89.
[45] O. S. Schneider, H. Seifi, S. Kashani, M. Chun, and K. E. MacLean, “Hapturk: Crowdsourcing affective ratings of vibrotactile icons,” in *Proc. of the CHI Conference on Human Factors in Computing Systems*, ser. CHI’16, 2016, pp. 3248–3260.
[46] M. O. Martinez *et al.*, “3-D printed haptic devices for educational applications,” in *Proc. IEEE Haptic Symp.*, 2016, pp. 126–133.
[47] C. Gallacher, A. Mohtat, and S. Ding, “Toward open-source portable haptic displays with visual-force-tactile feedback colocation,” in *Proc. Haptic Symp.*, 2016, pp. 65–71.
[48] J. Forsslund, M. Yip, and E.-L. Sallnäs, “WoodenHaptics,” in *Text Encoding Initiative’14*. New York, USA: ACM Press, Jan. 2015, pp. 133–140.
[49] P. Bucci *et al.*, “Sketching CuddleBits: Coupled prototyping of body and behaviour for an affective robot pet,” in *CHI’17*, 2017, pp. 3681–3692.
[50] A. Israr, S. Zhao, and K. E. A. McIntosh, “Stereohaptics: A haptic interaction toolkit for tangible virtual experiences,” in *SIGGRAPH’16*, 2016, pp. 13:1–13:57.
[51] K. Minamizawa, Y. Kakehi, M. Nakatani, S. Mihara, and S. Tachi, “TECHTILE Toolkit: A prototyping tool for design and education of haptic media,” in *Valley Ranch Islamic Center*. New York, USA: ACM Press, 2012, p. 2.
[52] C. Richard, A. Okamura, and M. Cutkosky, “Getting a feel for dynamics: Using haptic interface kits for teaching dynamics and controls,” in *Proc. ASME/IMECE Ann Symp Haptic Interfaces*, 1997, pp. 15–21.
[53] M. Wilson, “Six views of embodied cognition,” *Psychonomic Bulletin Rev.*, vol. 9, no. 4, pp. 625–636, 2002.
[54] Z. C. Zacharia, “Examining whether touch sensory feedback is necessary for science learning through experimentation: A literature review of two different lines of research across k-16,” *Educational Res. Rev.*, vol. 16, pp. 116–137, 2015.
[55] C. G. Rose, J. A. French, and M. K. O’Malley, “Design and characterization of a haptic paddle for dynamics education,” in *Proc. Haptic Symp.*, 2014, pp. 265–270.
[56] A. Otaran, O. Tokatli, and V. Patoglu, “Hands-on learning with a series elastic educational robot,” in *Proc. Intl Conf Human Haptic Sens. Touch Enabled Comput. Appl.*, 2016, pp. 3–16.
[57] PhET simulations. Accessed: June 2, 2019. [Online]. Available: https://phet.colorado.edu/en/simulations/category/new/
[58] C. Ramstein and V. Hayward, “The Pantograph: A large workspace haptic device for a multi-modal human-computer interaction,” in *CHI’94*. New York, New York, USA: ACM Press, pp. 57–58.
[59] Haply Robotics. Accessed: June 2, 2019. [Online]. Available: http://www.haply.co/
[60] K. E. MacLean and S. S. Snibbe, “An architecture for haptic control of media,” in *Proc. ASME/IMECE Ann Symp Haptic Interfaces Virtual Environ.*, vol. DSC-5B-3, 1999, pp. 219–228.
[61] Processing. Accessed: June 2, 2019. [Online]. Available: https://processing.org/
[62] Box2D: A 2d physics engine for games. Accessed: June 2, 2019. [Online]. Available: https://box2d.org/
[63] V. Braun and V. Clarke, “Using thematic analysis in psychology,” *Qualitative Res. Psychol.*, vol. 3, no. 2, pp. 77–101, 2006.
[64] “Thematic analysis,” Accessed on: Feb. 2019. [Online]. Available: www.psych.auckland.ac.nz/en/about/our-research/research-groups/thematic-analysis
[65] “QDA Miner Lite,” Accessed on: Dec. 2019. [Online]. Available: https://provalisresearch.com/products/qualitative-data-analysis-software/freeware/
[66] F. D. Davis, “Perceived usefulness, perceived ease of use, and user acceptance of information technology,” *MIS Quart.*, pp. 319–340, 1989.
[67] R. M. Felder and R. Brent, “Active learning: An introduction,” *ASQ Higher Educ. Brief*, vol. 2, no. 4, p. 1–5, 2009.
[68] “Introduction to haptics,” 2014–2018, Accessed on: Mar. 2019. [Online]. Available: lagunita.stanford.edu/courses/SelfPaced/Haptics/2014
[69] C. Swindells *et al.*, “Medium fidelity rapid prototyping of vibrotactile haptic, audio and video effects,” in *Proc. IEEE Haptic Symp.*, 2014, pp. 515–521.
[70] H. Culbertson, J. Unwin, and K. J. Kuchenbecker, “Modeling and rendering realistic textures from unconstrained tool-surface interactions.” *IEEE Trans. Haptics*, vol. 7, no. 3, pp. 381–93, Jul.–Sep. 2014.
[71] Stack overflow. Accessed: June 2, 2019. [Online]. Available: https://stackoverflow.com/

**Hasti Seifi** (Member, IEEE) received the B.Sc. degree from the University of Tehran, Tehran, Iran, in 2008, the M.Sc. degree from Simon Fraser University, Burnaby, BC, Canada, in 2011, and the Ph.D. degree in computer science from the University of British Columbia, Vancouver, BC, Canada, in 2017. She is currently a Postdoctoral Researcher with the Max Planck Institute for Intelligent Systems, Stuttgart, Germany. Her research interests are at the intersection of interaction design, haptics, and information visualization. She was the recipient of the Eurohaptics Best Ph.D. Thesis Award in the field of haptics in 2017, and an NSERC Postdoctoral Fellowship in 2018.



**Matthew Chun** received the B.A. degree in computer science from the University of British Columbia, Vancouver, BC, Canada, in 2016. He is currently working toward the M.Sc. degree in computer science (human computer interaction specialization) with the Sensory Perception and Interaction Research Group, University of British Columbia.



**Colin Gallacher** received the graduate diploma in computer science from Concordia University, Montreal, QC, Canada and the B.Eng. and M.Eng. degrees in mechanical engineering from McGill University, Montreal, QC, Canada. He has worked for the National Research Council of Canada, where he received the government Outstanding Achievement Prize for his work on neurosurgical haptic simulation tools, and later as a Research Associate in computer science with University of British Columbia. He is currently a research collaborator with CIRMMT, McGill University's Schulich School of Music and works on developing open-source haptic technologies as a founding member of the company Haply Robotics.

**Oliver Schneider** (Member, IEEE) received the BSc. Honours degree (most outstanding graduate in Computer Science) from the University of Saskatchewan, Saskatoon, SK, Canada, in 2010, the M.Sc. and Ph.D. degrees in computer science from the University of British Columbia, Vancouver, BC, Canada, in 2012 and 2016, respectively. From 2017 to 2018, he was a Postdoctoral Scholar with the Hasso Plattner Institute in Potsdam, Germany, and from 2014 to 2015 he collaborated with Disney Research. He has held NSERC Canada Graduate Scholarships (M.Sc. and Ph.D.) and a Postdoctoral Fellowship. He is currently an Assistant Professor of Human-Computer Interaction, Faculty of Engineering, Department of Management Science, University of Waterloo, Waterloo, ON, Canada .

**Karon E. MacLean** (Senior Member, IEEE) received the B.Scs. degree in biology and mechanical engineering from Stanford University, Stanford, CA, USA, the M.Sc. and Ph.D. degrees in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA. She has previously worked as a Robotics Engineer at University of Utah, Salt Lake City, UT, USA and as a haptics interaction researcher at Interval Research, Palo Alto. She is a Professor of Computer Science at the University of British Columbia, Vancouver, Canada. At UBC since 2000, she has been a connector of the haptics and HCI communities through conference chairing (HAPTICS, ACM UIST), journal co-founding (IEEE TRANSACTIONS ON HAPTICS), and cross-venue teaching and speaking. She has coauthored more than 100 peer-reviewed papers in computer science, engineering and psychology, and leads UBC's Designing for People interdisciplinary research cluster and graduate training program.