Haptic Experience Design

Tools, Techniques, and Process

by

Oliver Stirling Schneider

B.Sc. Honours, University of Saskatchewan, 2010

M.Sc., University of British Columbia, 2012

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

THE FACULTY OF GRADUATE STUDIES

(Computer Science)

The University Of British Columbia

(Vancouver)

August 2016

Abstract

Haptic technology, which engages the sense of touch, offers promising benefits for a variety of interactions including low-attention displays, emotional connections, and augmented media experiences. Despite an increasing presence of physical devices in commercial and research applications, there is still little support for the design of engaging haptic sensations. Previous literature has focused on the significant challenges of technological capabilities or physical realism rather than on supporting experience design.

In this dissertation, we study how to design, build, and evaluate interactive software to support haptic experience design (HaXD). We define HaXD and iteratively design three vibrotactile effect authoring tools, each a case study covering a different user population, vibrotactile device, and design challenge, and use them to observe specific aspects of HaXD with their target users. We make these in-depth findings more robust in two ways: generalizing results to a breadth of use cases with focused design projects, and grounding them with expert haptic designers through interviews and a workshop. Our findings 1) describe HaXD, including processes, strategies, and challenges; and 2) present guidelines on designing, building, and evaluating interactive software that facillitates HaXD.

When characterizing HaXD processes, strategies, and challenges, we show that experience design is already practiced with haptic technology, but faces unique considerations compared to other modalities. We identify four design activities that must be explicitly supported: *sketching*, *refining*, *browsing*, and *sharing*. We find and develop strategies to accommodate the wide variety of haptic devices. We encapsulate approaches for designing meaning with haptic experiences, and finally, highlight a need for supporting adaptable interfaces.

When informing the design, implementation, and evaluation of HaXD tools, we discover critical features, including a need for improved online deployment and community support. We present steps to developing research software into mature a HaXD suite of tools, and reflect upon evaluation methods. By characterizing HaXD and informing supportive tools, we make a first step towards establishing HaXD as its own field, akin to graphic and sound design.

Revision: r0.4

Preface

No creative work occurs by a lone individual; this dissertation is no exception. All of the projects described in this work are collaborative efforts in at least some capacity. Even where the author contributed all work, there was often informal feedback from friends, family, and colleagues. As such, this dissertation will use the first-person plural, "we", throughout. In this preface, we clarify the author's contribution to the work, much of which has been published.

In Chapters 1, 2, and 9, Oliver contributed writing and framing, with feedback provided by the supervisor (Dr. Karon MacLean) and supervisory committee (Drs. Ronald Garcia and Michiel van de Panne) throughout his PhD program. Some of this thinking ⌈**OS** *where?*⌋ is combined with a handbook chapter currently under review, written with Dr. MacLean as lead other, with Oliver and PhD candidate Hasti Seifi as co-authors. This chapter is aimed as an advanced (*i.e.*, graduate or senior undergraduate) educational resource incorporating Oliver and Hasti's research.

In Chapter 3, Oliver contributed all work and ideas, with feedback and guidance from supervisor Dr. Karon MacLean. The software has been released as an open-source project at https://github.com/ubcspin/mHIVE. This work has been published as full conference paper with an associated demo at HAPTICS'14, and at a workshop at CHI'14:

> Schneider and MacLean. (2014) *Improvising Design with a Haptic Instrument*. HAPTICS '14.
>
> Schneider and MacLean. (2014) *mHIVE: A WYFIWIF design tool*. 2014 IEEE Haptics Symposium (HAPTICS).
>
> Schneider and MacLean. (2014) *Reflections on a WYFIWIF Design Tool*. Workshop on Tactile User Experience Evaluation Methods at CHI 2014.

In Chapter 4, Oliver contributed most work and ideas, with initial interviews with designers and haptic experts conducted by Disney Research. This work was conducted while on internship at Disney Research Pittsburgh, with some supplementary work done at UBC. Dr. Ali Israr supervised Oliver's internship; Oliver led writing with feedback and guidance from Drs. Israr and MacLean. This work was presented by Oliver at UIST'15 with an associated demo:

> Schneider, Israr, and MacLean. (2015) *Tactile Animation by Direct Manipulation of Grid Displays*. UIST'15.
>
> Schneider, Israr, and MacLean. (2015) *Tactile Animation by Direct Manipulation of Grid Displays*. UIST '15 Demos.

In Chapter 5, Oliver contributed all work and ideas, with feedback and guidance from Dr. MacLean. Macaron has been released as an open-source project at https://github.com/ubcspin/Macaron and is available online at http://hapticdesign.github.io/macaron. Subsequent development of the core Macaron tool and extension MacaronMix includes work by Matthew Chun, Benson Li, Ben Clark, and Paul Bucci. The study reported in Chapter 5 was presented by Oliver at HAPTICS'16 with an associated demo:

> Schneider and MacLean. (2016) *Studying Design Process and Example Use with Macaron, a Web-based Vibrotactile Effect Editor*. HAPTICS '16: Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems.
>
> Schneider and MacLean. (2016) *Macaron: An Online, Open-Source, Haptic Editor*. HAPTICS '16 Demos (finalist for best demo).

In Chapter 6, Oliver was part of a collaborative team together with PhD candidate Hasti Seifi, undergraduate summer student Matthew Chun, and master's student Salma Kashani, all supervised by Dr. MacLean. Oliver and Hasti planned and managed the project, with Matthew and Salma doing proxy design, study design, and data collection for low-fidelity proxies and visual proxies respectively. Oliver lead paper writing and quantitative analysis, working closely with the other authors, and presented the work at CHI'16:

> Schneider, Seifi, Kashani, Chun, and MacLean. (2016) *HapTurk: Crowdsourcing Affective Ratings for Vibrotactile Icons*. Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16.

Chapter 7 describes several focused projects to give this dissertation improved breadth. Oliver played different roles depending on the project.

Section 7.1, FeelCraft Oliver worked closely with Siyan Zhao, supervised by Dr. Israr at Disney Research Pittsburgh. Oliver implemented the rendering system (co-developed with the engine described in Chapter 4), developed the MineCraft plugin and connection architecture, and wrote the AsiaHaptics paper (archived in LNEE 277) with feedback from Ali Israr. Artistic contributions to the video were made by Kyna McIntosh and Madeleine Varner. Oliver and Siyan together designed the implemented feel effects (Oliver lead implementation), planned, shot, and edited the video submissions (Siyan lead editing); each presented the demo once (Oliver at AsiaHaptics 2014, Siyan at UIST 2014):

> Schneider, Zhao, and Israr. (2015) *FeelCraft: User-Crafted Tactile Content*. Lecture Notes in Electrical Engineering 277: Haptic Interaction.
>
> Zhao, Schneider, Klatzky, Lehman, and Israr. (2014) *FeelCraft: Crafting Tactile Experiences for Media using a Feel Effect Library*. UIST '14 Demos.

Section 7.2, Feel Messenger Oliver worked closely with Siyan Zhao and Dr. Israr. All three developed the concept. Siyan lead poster design and assisted with figures. Dr. Israr lead writing assisted by Oliver and Siyan, and presented this work at CHI'15. Oliver designed and implemented the Feel Messenger application, conducted part of the preliminary study, and lead the demo submission and presentation at World Haptics 2015:

> Israr, Zhao, and Schneider. (2015) *Exploring Embedded Haptics for Social Networking and Interactions*. Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '15.
>
> Schneider, Zhao, and Israr. (2015) *Feel Messenger: Embedded Haptics for Social Networking*. World Haptics '15 Demos.

Section 7.3, RoughSketch Oliver was the senior graduate student on a four-person student team including Paul Bucci, Gordon Minaker, and Brenna Li. All four contributed ideas and haptic designs and iteratively developed the final submission. Paul lead graphic design efforts; Gordon and Brenna presented the work at World Haptics 2015. RoughSketch won first place among 10 finalists.

Section 7.4, HandsOn Oliver helped supervise Gordon Minaker during a summer NSERC placement and directed studies, with Dr. MacLean supervising and PhD student Richard Davis collaborating. This work was part of a larger collaborative effort including Melisa Orta Martinez, Dr. Allison Okamura, and Dr. Paulo Blikstein from Stanford University. Gordon lead the system design and implementation, study design, facilitation, and analysis, and paper writing and submission. Oliver helped supervise Gordon throughout this process, assisted and supervised by Dr. MacLean. Richard helped plan the study, implement software, write the paper, and provide insights for study implementation. All three assisted with poster design. Dr. MacLean presented the work at EuroHaptics 2016; the system was also included in a demo presented by Melisa at HAPTICS'16:

> Minaker, Schneider, Davis, and MacLean. (2016) *HandsOn: Enabling Embodied, Creative STEM e-learning with Programming-Free Force Feedback*. Haptics: Perception, Devices, Control, and Applications: 10th International Conference, EuroHaptics 2016, London, UK, July 4-7, 2016, Proceedings, Part II.

> Martinez, Minaker Gordon, Davis, Schneider, Morimoto, Taylor, Barron, MacLean, Blikstein, and Okamura. (2016) *HandsOn with Hapkit 3.0: a creative STEM e-learning framework*. HAPTICS '16 Demos.

Section 7.5, CuddleBit Design Tools Oliver collaborated closely with undergraduate David Marino and master's student Paul Bucci, supervised by Dr. MacLean and with support from Hasti Seifi. Oliver supervised David through his directed studies project, and helped worked with Paul and David in developing and designing the Voodle system. Oliver worked with Paul Bucci to extend Macaron into MacaronBit and contributed writing to a demo presented at EuroHaptics 2016 by Dr. MacLean:

> Bucci, Cang, Chun, Marino, Schneider, Seifi, and MacLean. (2016) *CuddleBits: an iterative prototyping platform for complex haptic display*. EuroHaptics '16 Demos.

In Chapter 8, UBC alumnus Dr. Colin Swindells conducted interviews and developed interview notes and initial analysis ideas in 2012, supervised by Dr. MacLean and Dr. Kellogg Booth. In 2015-2016, Oliver transcribed and analyzed the collected interviews, organized and analyzed the HaXD'15 workshop (oliverschneider.ca/HaXD/) with guidance from Dr. MacLean, and lead writing of a manuscript. Drs. MacLean and Booth contributed to writing; Dr. Swindells provided feedback.

Because much of this work has been peer-reviewed, we reproduce published papers as chapters in this dissertation. Chapters 3-6 and 8 each include a newly-written preface to introduce the work, then includes the corresponding paper with only minor formatting modifications. In this way, we preserve the original argumentation of each published work while connecting it to this dissertation's overall goals and findings.

Contents

List of Tables

List of Figures

Acknowledgments

Thank those people who helped you.

Don't forget your parents or loved ones.

You may wish to acknowledge your funding sources.

Chapter 1

Introduction

Technology changes. Symbolic, machine-focused communication like punch cards, assembly languages, and terminal interfaces yield to natural, physical, always-connected interactive systems. The emergence of virtual and augmented reality (VR & AR), rapid development of personal fabrication techniques, and explosion of wearable and cyberphysical technologies propel us towards a mixed physical-digital world at an accelerating pace. As computers expand beyond screens and keyboards, we look to engage the rich senses of touch.

Haptic experiences are moving from niche roles to mainstream adoption. Haptic technology includes both the tactile (skin-based) and proprioceptive (force- and position-based) components of touch. Recently, other natural interaction techniques like touchscreens and voice control held the limelight, with haptic feedback relegated to buzzing alerts or limited to high-stakes expert systems like laparoscopic surgery. Now, new media seeks deeper immersion, smart environments look to connect physically with users, and consumer devices like the Apple Watch and Pebble adopt high-fidelity haptic actuators. The question is how to enable people to craft experiences with these technologies.

The diverse field of haptics has actively engineered new devices and studied human perception, but the design of haptic experiences remains a critical challenge. Little is known about this nascent field of design, with many unique challenges: haptic experiences are rich, diverse, multimodal entities which necessitate in-person interaction and have limited infrastructure. How can we support creativity with these experiences, empowering artists, developers, designers, and scientists to effectively work with this emerging medium? In this dissertation, we study the process of haptic experience design (HaXD) and establish guidelines for building interactive software systems to support it.

1.1 Haptic Experience Design (HaXD)

We define HaXD as:

> *The design (planning, development, and evaluation) of user experiences deliberately connecting interactive technology to one or more perceived senses of touch, possibly as part of a multimodal or multisensory experience.*[1]

We use HaXD instead of "haptic design", which can also refer to design practices related to haptics but not directly involving the user experience, *e.g.*, mechanical design of a new actuator. We use "haptic designers" or "hapticians" to refer to haptic *experience* designers (those who practice HaXD).

In this dissertation, we take a systems approach to design. Designers do not exist in a vacuum, but rather in in a physical, personal, social, and cultural context. We adopt a framework of design activities which are practiced in general experience design, but need explicit support in HaXD. Our research identified four activities (Figure 1.1): *sketching*, ad-hoc, suggestive exploration; *refining*, iteration and fine-tuning; *browsing* examples and drawing from experience; and *sharing* designs for feedback and posterity. Chapters 3-6 explore these activities directly.

1.2 Why is HaXD hard?

HaXD faces two types of challenges: those resulting from its relative youth, and those intrinsic to the sense of touch and touch-based technology. One of the goals of this dissertation is to articulate the challenges that have been known informally for years, and capture those that are not as well known.

Haptic experience design is a young field: the first conference to explicitly focus on haptics was Haptics Symposium in 1992. As a result, design for haptic experiences is not as mature as with vision and audio, which can draw from centuries of music and graphic design, and decades of sound design, which have translated well to their digital equivalents. We see this with limited infrastructure to support the wide variety of haptic devices [95], to the point where online distribution is current research [1], and with limited, varied language for touch [111].

There are also intrinsic challenges when designing for touch: variabilities in low-level perception due to, *e.g.*, individual differences [138], device location and user activity [121], and aging [212, 213]; a strong influence of user preferences [200, 201]; complexity of the haptic senses [43, 119, 127]; and tight technical constraints cross-cutting software and hardware [95, 135]. Throughout this work, we identify and characterize these challenges, and make progress to conquer them.

[1] We developed this definition from our interviews with hapticians (Chapter 8).

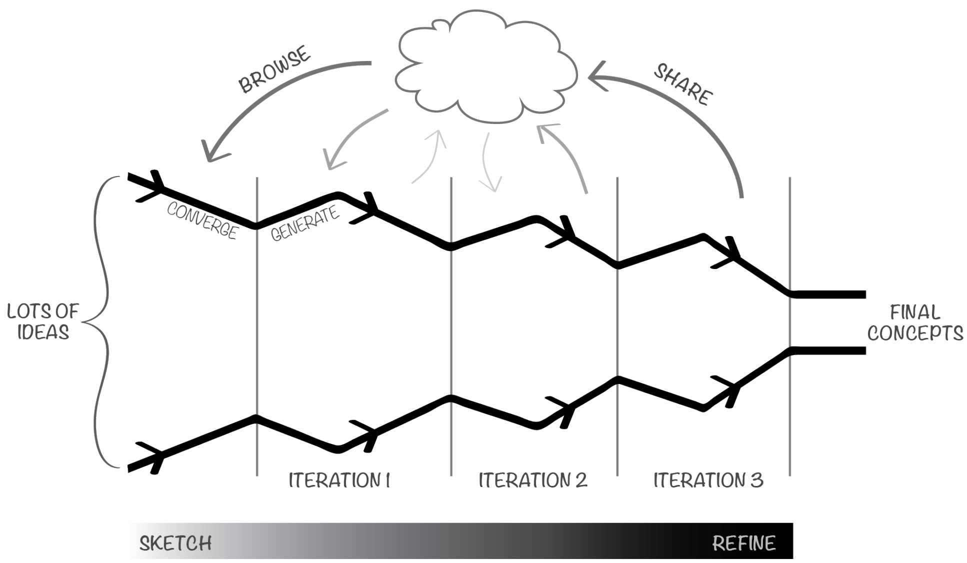


Figure 1.1: An adapted version of the classic design funnel, where multiple initial ideas are iteratively developed into final concepts [27]. We add four design activities that we found need to be explicitly supported for HaXD: *sketching*, *refining*, *browsing*, and *sharing*.

1.3 Approach

We approach this problem with three strategies: vibrotactile case studies for depth, several focused design projects for breadth, and data from haptic designers to ground our findings (Figure 1.2).

1.3.1 Depth: Vibrotactile design tool case studies (Chapters 3-6)

To understand design, we take a design perspective. In each of three case studies, we design, build, and evaluate a tool or technique to support an aspect of HaXD, scoped to *vibrotactile* (VT) design. Each of these results in concrete implications for designing tools and a small window onto the larger HaXD process. Contributions include algorithms, data structures, interaction techniques, features, analytic techniques, and working software tools that have been employed by designers. Chapter 3, Chapter 4, and Chapter 5 outline iterative development and evaluation of VT design tools; Chapter 6 covers a VT design technique (proxies).

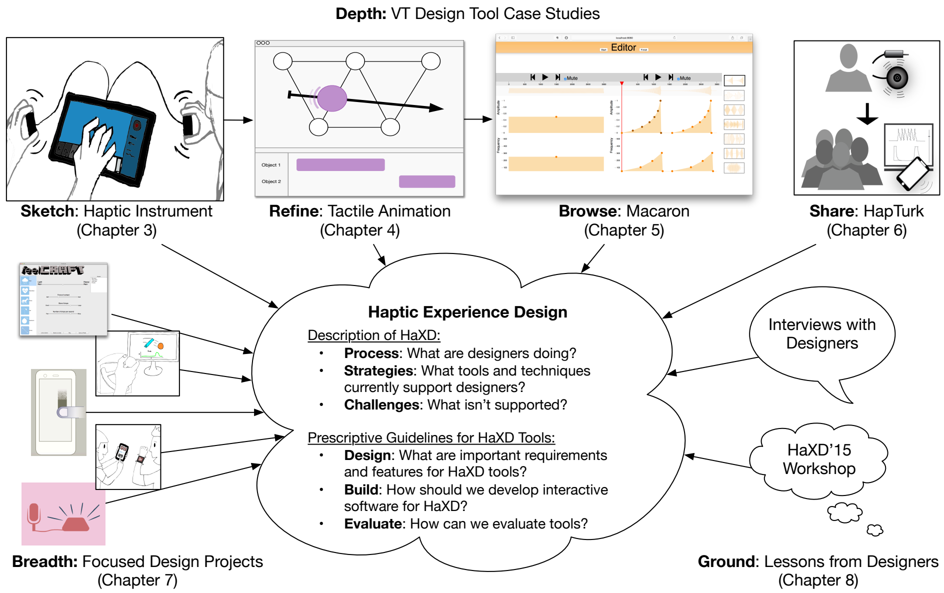


Figure 1.2: Approach overview. We investigate VT design tools (Chapters 3-5) and techniques (Chapter 6) in-depth. These findings are synthesized with multiple, smaller focused projects (Chapter 7) and grounded data from hapticians (Chapter 8) into a preliminary understanding of HaXD.

1.3.2 Breadth: Focused haptic design projects (Chapter 7)

While the case studies provide an in-depth investigation into VT sensation design, results may not generalize to other devices, and provide limited investigation into application areas like education. To generalize from VT effects, explore other aspects of haptic design, and gain personal experience as haptic experience designers, we participate in several smaller focused design projects, which lend a broader context to our findings. Chapter 7 discusses these projects.

1.3.3 Ground: Data from haptic experience designers (Chapter 8)

Finally, despite the recent growth of the field, haptic designers remain relatively rare and difficult to recruit. To complement our primarily design-based approach and ground it with haptic experience designers in the field, we draw from other data sources: a workshop held at World Haptics 2015 and interviews with haptic designers. Chapter 8 discusses this characterization of HaXD, and serves as a capstone by defining HaXD and articulating a vision for how HaXD might manifest.

1.4 Outline and Contributions

This dissertation continues as follows. First, in Chapter 2, we present the necessary background with an overview of haptic technology and perception, the value of haptics and related applications, design theory from non-haptic fields, existing haptic design tools and techniques, and underlying methodology of our work.

Then, we outline each VT case study in Chapters 3-6. In Chapter 3, we present findings from our first vibrotactile design tool, the haptic instrument, which supported easy exploration and informal feedback (*sketching*), but identified a key problem: it did not support *refining* designs. In Chapter 4, we present findings from our second vibrotactile design tool, Mango, which established a generalized pipeline and was able to support both *sketching* and *refining* for expert visual animators; it highlighted reuse as an important next step. In Chapter 5, we present findings from our third vibrotactile design tool, Macaron, which implemented a *browsing* interface and analytics system; we found examples played a large part of the design process, and used user logs to provide a picture of our participants' design process, including confirmation of project preparation and *browsing*, initial design, *sketching*, and *refining*. In Chapter 6, we document findings from HapTurk, a technique for *sharing* vibrotactile designs for feedback at scale: from the crowd using proxy vibrations distributed over Mechanical Turk.

We then describe focused haptic design projects in Chapter 7, and the results from our grounded data collection in Chapter 8. In Chapter 7, we synthesize together findings from our side projects, showing generality by applying our understanding of haptic design explicitly in several domains and gaining practical experience designing haptic experience. In Chapter 8, we complement our design-based inquiry through interviews with professional haptic designers and a workshop run to elicit feedback from the community; this captures a description of haptic design, reinforcing our findings for important support tools, and identifies more systematic challenges.

Finally, in Chapter 9, we conclude with a synthesis of our final results and directions for future research.

Chapter 2

Background

In this chapter, we provide the relevant background for this dissertation. We begin with an overview of haptic technology and perception. Next, we discuss the application space for haptics and why haptic experiences are increasingly important to design. We then discuss non-haptic creativity support tools and design theory which provided inspiration and guiding principles. After, we discuss the previous work in HaXD and related support tools, identifying why this is an area for improved understanding. Finally, we present the qualitative and quantitative methodologies used in this dissertation. Throughout the chapter, we contextualize this work and HaXD in both the haptics and HCI communities.

2.1 An Overview of Haptic Technology and Perception

The term "haptic" was coined by German researcher Max Dessoir to refer to the study of touch, in a similar to "optic" for sight and "acoustic" for sound [84]. Today, it refers to both the study of the psychology and perception of the senses of touch, and the technology that employs touch as a method of feedback. Haptic technology is typically separated into two classes based on the main sense modality: *tactile* (or *cutaneous*) sensations, and *proprioception*, or the sense of body location and force; the latter includes *kinaesthetic* senses of force and motion. These two types of feedback are useful for different purposes, *e.g.*, people use their fingerpad's tactile senses to derive texture, but kinaesthetic feedback to infer weight [126]; different senses can be combined for more convincing results [165]. For an overview of the haptic senses, we direct the reader to Lederman and Klatzky [127]; for a practical introduction to haptic technology, we suggest Hayward and Maclean [95]. We focus our coverage on the sensations directly studied by this dissertation, while also portraying the diversity of haptic experiences and technology.

2.1.1 Tactile Perception and Technology

Tactile sensations rely on multiple sensory organs in the skin, each of which detect different properties, *e.g.*, Merkel disks detect pressure or fine details, Meissner corpuscles detect fast, light sensations (flutter), Ruffini endings detect stretch, and Pacinian corpuscles detect vibration [43]. Of these, the Pacinian corpuscle is most widely targeted by technology through vibrotactile (VT) sensations, where vibrations stimulate the skin. VT sensations are accessible, well-studied, and increasingly widespread, and can be passively felt, easing implementation. Our in-depth design tool studies thus focus on VT experiences.

VT actuators can take may forms. Eccentric mass motors (sometimes "rumble motors" or "pager motors") are found in many mobile devices and game controllers, and are affordable but inexpressive. More expressive mechanisms such as voice coils offer independent control of two degrees of freedom, frequency and amplitude. Piezo actuation is a very responsive technique that is typically more expensive than other vibrotactile technology. While voice coils typically directly stimulate the skin, linear resonant actuators (LRAs) shake a mass back and forth to vibrate a handset in an expressive way; a common research example is the Haptuator [233]. Instead of directly stimulating the skin, this actuator typically shakes another device held by the user, such as a mobile device [240] or pen [51]. As of 2016, LRAs are increasingly deployed in consumer products (*e.g.*, Apple's Taptic Engine).

Actuators like VT devices can be used alone or put together spatial multiactuator displays like seats [104, 107], belts [169, 173], wristbands [5, 89, 169], vests [115, 174], and gloves [123, 171]. These can be arrange into grids, either dense tactile pixels ("taxels") [123] or sparse arrays [104, 107], to provide 2D output on a plane. Multiactuator arrays increasingly exploit tactile illusions to create effects of motion or phantom sensations in-between actuators.

Another emerging tactile feedback mechanism is programmable friction. Surface friction, for example on a mobile touch screen, can be manipulated by both mechanical motion or electrical adhesion. The TPad [232] vibrates a plate at ultrasonic frequencies to create a cushion of air between the surface and the user's finger. This effect is programmable, and can be used to with a number of interactive scenarios [134]. Other techniques like electrovibration, deployed in TeslaTouch [10], and electrostatic forces [151], can create a similar effect. Strong electroadhesion [207] has the potential to create very large shear forces, but comes with a high power cost. In RoughSketch (Chapter 7), we design for a mobile version of the TPad deployed on Android devices, the TPad phone (www.thetpadphone.com).

There are many other types of tactile stimulation used in haptic experiences. 2-dimensional pin-based grids like Optacon [11] and HyperBraille (www.hyperbraille.

de) can display Braille and 2D images to the blind and visually impaired, and can operate as a generic computer display [175]. Similar multi-point displays have been deployed on mobile devices. Edge Haptics uses dozens of linearly-actuated pins on the edge of a mobile device for tactile stimulation, similar to a 2-dimensional braille pin display [110], while laterally moving pins can use skin-stretch as a display mechanism [139]. Electrocutaneous stimulation, where electrodes directly stimulate the skin, has been deployed for spatial tongue displays [8]. Temperature displays exploit warm and cold receptors in thea skin for display, using Peltier junctions [118]. Tactile sensations can be created at a distance using ultrasonic transducers [32, 162] and vortex cannons that shoot puffs of air [210].

2.1.2 Proprioception and Force Feedback

Proprioception, the sense of force and position, is synthesized from multiple sensors as well: the muscle spindle (embedded in muscles), golgi-tendon organ (GTO) in tendons, and tactile and visual cues [119]. We distinguish proprioception from the related term kinaesthetic by being the general, synthesized sense, where kinaesthetic sensation is strictly the sense of motion ⌈**OS** *KM: make sure this is correct*⌋. Force displays are common in precise, specialized applications like robot-assisted surgery [166] or realistic sensorimotor training environments [224].

Force-feedback devices might have degrees of freedom of feedback (DoF), the number of forces or torques they can display. These devices render a *virtual environment*, with simulated forces depending on the input from the user. Common consumer-facing 3-DoF devices include the Geomagic Touch (previously the Sensable PHANTOM) and Falcon devices, offering force in three directions. 2-DoF designs like the pantograph [29, 177] can provide displays on screens, walls, and tables. Research with these displays often requires realistic simulation and rendering: *e.g.*, making free space feel free, providing stiff virtual objects and walls, and avoiding saturation [149]. Open-hardware, self-assembled versions of these devices, such as WoodenHaptics [74] for 3-DoF devices and Haplet [77] for 2-DoF displays, have the potential to make haptics more accessible. Much previous work has been done on handling technical concerns, *e.g.*, displaying complex polygonal objects with a "God object" [243], coordinating remotely situated devices or shared environments [26], and improving collision realism with transient forces [124]. More complex environments are primarily programmed in using APIs like CHAI3D, OpenHaptics, or Unity.

Another approach is to use simple force feedback, especially for haptics education [114]. 1-DoF devices include linear actuators pushing on the user and haptic knobs, *e.g.*, the UBC Twiddler [67, 144, 204], and paddles, *e.g.*, the HapKit [168]. The UBC SPIN lab has also adopted 1-DoF force feedback in its affective robot, the

Haptic Creature [236, 237], the CuddleBot [2], and CuddleBits [30]. We explore force-feedback design with the HapKit and CuddleBits in Chapter 7.

2.1.3 Haptic Illusions

Like the stroboscopic effect transforming a series of images into the perception of motion for visual displays, illusions play a valuable role in haptic sensations [94]. Some effects are influenced by other senses. In the classic size-weight illusion [40], when two weights have the same mass but different sizes, the smaller is perceived to be heavier, whether size is seen or felt [94]. A striking, recent example is the use of visual dominance to use a single physical block to provide haptic feedback for multiple virtual blocks by distorting the visual position of the user's arm [7]. We employ similar techniques in our FeelCraft and Feel Messenger projects, using visual feedback to prime users to haptic sensations (Chapter 7).

Other illusions are purely tactile and useful for multiactuator displays. Phantom tactile sensations [3], create illusory vibrations in between two or more VT actuators, opening up the space in-between actuators for display. Continuous motion can be simulated, *e.g.*, Seo and Choi [202] created a perceived motion flow between two VT actuators mounted on the ends of a handheld device by controlling their intensity. Similarly, Lee et al. [133] created across-the-body and out-of-the-body illusions on a mobile device using up to four LRAs; Gupta et al. [89] used interpolation on a VT wristband for new interaction techniques. The Tactile Brush algorithm [105] combined phantom tactile sensations and apparent tactile motion to render high-resolution and moving haptic patterns on the back using a coarse grid of VT actuators. Other spatio-temporal VT illusions such as the "cutaneous rabbit" [218], where carefully timed discrete tactile stimuli create perceived motion, and Tau and Kappa effects [93, 94], where perceived distance between stimuli depending on their timing, can also be used with VT arrays. Similar illusions are possible using other tactile modalities, including temperature displays [208] and electrocutaneous stimulation [220]. We extend phantom VT sensations to 2D interpolation (*e.g.*, between 3 actuators) to enable Tactile Animation (Chapter 4).

Of course, haptic perception can depend on the user's physical and attentional connection with the device, especially important in wearable contexts. Vibrotactile detection depends on many variables, including location on the user's body, how much the user is moving, and whether they are expecting the vibration [121], and social context [33]. These effects can be mitigated through sensing, *e.g.*, detecting movement with accelerometers [13]. The implications of context on HaXD are discussed by professional designers in Chapter 8.

2.2 The Value of Haptic Experiences

Haptic feedback can provide several benefits to interactive experiences. Here, we outline the main benefits haptics provides, and then several application areas that commonly leverage those benefits.

2.2.1 Why Touch?

Haptic technology enables information transfer between humans and computers; this transfer is rich, proximal, and fast. Information flows both ways, through input and output, sometimes simultaneously. We focus on designed haptic display.

One advantage of touch is simply that it is not vision or audio, the primary feedback methods for interactive systems. Haptic technology can reinforce other modalities, enriching feedback for a more complete experience, or provide complementary feedback, with many possible reasons: information saturation, *e.g.*, when visual or audio displays have maximized their output; task context, *e.g.*, when the user is driving and must keep their eyes on the road; impairment or impairing situations, *e.g.*, when a user has limited sight or hearing; ambient displays, *e.g.*, keeping a user aware of a piece of information without interrupting them; or nature of the information, *e.g.*, communicating emotion. Sensory substitution, first pioneered by Bach-y-Rita [9], is a dramatic technique often using haptic senses to augment or replace other senses. A wide variety of devices have been developed and studied for the visually impaired (*e.g.*, [11, 175]).

Of course, touch is a unique, rich sense in its own right. Like sound, touch can be invisible; like vision, it can be spatial. Feeling an object is especially helpful at discerning material properties [126]. Touch is the first sense to develop, playing an important role in formative experiences [111]; sensorimotor actions can help to scaffold understanding through embodied learning [170]. Touch can also be used for artistic expression; Gunther et al. [88] studied a full-body vibrotactile suit to create music-like “cutaneous grooves”, helping to identify the artistic space of VT sensations, including concerts with tactile compositions.

While haptic feedback can improve usability and task performance [35, 173], touch is especially connected to visceral, emotional connections. Marketing research has studied multiple ways that touch can connect with customers: the way a smartphone feels can influence a purchase over an alternative that might work better, and customers prefer to shop at stores that let them touch products [111, 211].

To study emotion and technology, researchers commonly draw from two affective models: Ekman’s basic emotions and Russell’s affect grid. Ekman’s basic emotions [64, 65] are a discrete set of emotions identified from a cross-cultural study of facial expressions; we use this model’s emotions as the design task in

Chapter 3. Russell's affect grid [181, 182] separates emotions into dimensions of arousal (low to high energy) and valence (positive and negative emotions); this work informs much of our work on expressivity and especially the CuddleBit work in Section 7.5.

Researchers are starting to develop design guidelines to express emotions through haptic experiences. Low-level parameters like amplitude, frequency, and duration have been linked to emotions: Yongjae Yoo et al. [239] showed that VT icons can express arousal and valence; Obrist et al. [163] established design parameters for mid-air ultrasound stimulation. Because touch can be bidirectional, affective sensing can accompany haptic display. The Haptic Creature project established a touch dictionary of gestures used to emotionally communicate with robots [237]. Touch-based surfaces can detect these gestures [70] through technologies like conductive fur and fabric [71].

2.2.2 Applications

While realistic virtual environments for force-feedback haptic feedback is helpful in medical or training applications [166, 224], we focus on applications that find increased value in an explicit design step.

Immersion

A popular application for haptic experiences is augmented, immersive media experiences. Actuated tactile feedback has been used as early as 1959 in the movie *The Tingler* [103]. 4D theatres and theme park rides use bursts of air or water sprays to engage the audience. Companies like D-Box (www.d-box.com) augment films with haptic tracks that both low-frequency movements and high-frequency vibrations, and can be found in theatres across the world. Buttkicker (www.thebuttkicker.com) also augments 4D theatres, and provides products for home theatre setups.

Haptic experiences are also increasingly of interest in virtual reality (VR) environments. Skin stretch techniques, explored in [87] and now commercialized by Tactical Haptics (tacticalhaptics.com), augments virtual-reality setups by simulating forces and torques using handheld controllers, lending stronger immersion for virtual environments and VR games. Haptic Turk [41] and TurkDeck [42] are innovative explorations of high-fidelity haptic experiences in virtual environments using people as actuators. Impacto uses electrical muscle stimulation and a solenoid actuator to create wearable haptic feedback with both kinaesthetic and tactile feedback [137]. Haptic retargeting distorts visual feedback to re-use a single physical block in a virtual block-building game [7].

Previous work has also attempted to add greater immersion to broadcast media by including haptic sensations. Modhrain and Oakley [153] present an early vision of Touch TV, using active touch with two-DOF actuators embedded in remote controllers; Gaw et al. [79] follow up with editable position playback on a force-feedback device, played alongside movies or cartoons. More recently, the proliferation of online streaming video has developed opportunities to add haptic sensations using novel data formats. Researchers have looked at how to integrate a haptic track into Tactile Movies [123], YouTube [78], or haptic-audiovisual (HAV) content [56], complete with compositional guidelines drawing inspiration from film and animation [86].

Affect

We've discussed how touch is closely connected to emotion. This has implications for design; for example, couples are more comfortable with a "hand stroke" metaphor for two remotely coupled haptic devices than strangers, who prefer a more less intimate "ping-pong" metaphor [209]. Emotional display through touch has therapeutic applications. Bonanni et al. [15] created TapTap, a wearable that can record and playback VT equivalents of affectionate touch to support users in therapy. Tactile displays target improved mental health [225] and aiding emotional understanding for autism [39]. The Haptic Creature project explores affective touch in human-robot interaction (HRI) [235–238]; this furry, zoomorphic robot can measurably relax users when they feel it breath [199].

Communication

Touch is extremely important for interpersonal communication, from greeting a new acquaintance with firm handshake, to showing affection to a loved one; see Gallace [76] for an overview. Of course, technologically can mediate touch between people, *e.g.*, in remote collaboration or shared virtual environments [90]. Brave and Dahley [16] introduced "inTouch", mechanically linked rollers that enabled playful touch interactions at a distance. ComTouch [37] used pressure input to send vibrations with between mobile phones, finding it was used for attention, turn-taking, and emphasis. Hoggan et al. [98] elaborated these findings a one month-study found users sent "Pressages" (pressure messages) both for greetings and to emphasize speech or emotional messages. Chan et al. [35] used VT icons to coordinate turn-taking in an online system, featuring an extensive design process to create and perceptually verify icons that present system state and requests with varying urgency.

Notification

Mobile contexts are rife with opportunities for haptic feedback. Ambient tactile displays can provide awareness can provide awareness and alerts without distracting the user. VT feedback is affordable, low-power, and can be added to watches and wrist-bands [5, 24], belts, vests, and other wearables. Tactons [22] are a type of haptic icon [140] that provide VT feedback, commonly in mobile applications. Rhythm opens up a large design space, letting users learn 84 different icons [221] and can be applied with even light, low-cost rumble motors. A 28-day study showed that rhythmic VT icons do not disturb users in daily activities and can communicate ambient information [33]. Hemmert and Joost [96] explored a life-like metaphor of pulsing and breathing to provide alerts, but found care needed to be taken to not be annoying. Multiple actuators can be combined in mobile hand-held devices to provide differentiable spatial information, enriching the VT icon design space [234]. VT icons produced by phones can represent multiple levels of urgency and source of an alert (*e.g.*, voice call, text message, or multimedia message) [21].

Guidance

Guidance is a typical application for VT feedback, which can be invisible, mobile, and accessed without using vision or sound. Spatial guidance through haptic wearable display can improve navigation with multiple actuators across several form factors, including belts [136, 173], wrist-bands [5], and vests [174]; in each case, the vibrations inform the user where to go with spatial vibrations or metaphorical spatial icons. Periodic vibrations can guide a user's walking speed without large attentional demands [120]. Tactile illusions like saltation can provide directional information for guidance [219]; larger back-based displays are effective for guiding both attention and direction, *e.g.*, in automobiles [217]. Brewster and Constantin [18] used VT icons to provide awareness of nearby friends and colleagues.

Education

Haptic technology has the potential to improve educational resources, especially to those lacking resources. Montessori methods have long espoused the value of physical learning aids, especially using physical *manipulatives* [154]. There is evidence to support these techniques: in a meta-analysis of 55 studies, Carbonneau et al. [31] found that physical manipulations improve several learning outcomes, with influence by other instructional variables. Studies of gestures have also found value in students "being the graph" by physically acting out mathematical shapes, grounding abstract knowledge in embodied experience [80]. These techniques have roots in constructivist learning, where learners use existing understandings as a *transitionary object* to understand new concepts [170].

Haptic technology is well-positioned to support embodied learning, and there is early evidence for its efficacy. Haptic feedback has been shown to improve temporal aspects when training motor skills [69]. In a study for molecular chemistry education, Sato et al. [185] found students had higher test scores when they interacted with their haptic learning interface; students reported engagement. In Chapter 7 we describe results from an early learning interface for low-cost haptic displays [168], showing that haptic technology can improve engagement and make lasting impressions.

2.3 Non-Haptic Design and Creativity Tools

Design has an increasingly studied topic in non-haptic contexts, providing both guidelines and inspiration for HaXD. In this section, we present related work on non-haptic design organized into three major elements: problem preparation, hands-on design, and collaboration.

2.3.1 Problem Preparation

Creative tasks, like design, are often defined as the recombination of existing ideas, with a twist of novelty or spark of innovation by the individual creator [227]. Also known as the "problem setting" [198], "analysis of problem" [227], or "collect" [205] step, problem preparation involves getting a handle on the problem, drawing inspiration from previous work. Schön demonstrated that designers initially frame their problems before developing a solution [198]. Schön also describes the designer's repertoire, their collected experience, which aids in design. External examples are especially useful for inspiration and aiding initial design [27, 97]. In our work, the design activity of *browse* overlaps significantly with problem preparation.

2.3.2 Hands-On Design

There has recently been a shift in how we interpret the act of thinking. No longer is thinking relegated to the head; cognition is now seen as being situated in the physical world [100]. The designer must iteratively generate a varied set of initial ideas (ideation) and then prune them (evaluation), repeating this step many times to settle on a single design [27]. Working with multiple ideas simultaneously is a boon to good design. Developing interfaces in parallel can facilitate generation and evaluation, delaying commitment to a single design [92, 178], while in groups, sharing multiple designs improves variety and quality of designs [60].

Sketching supports ideation, evaluation, and multiple ideas, allowing the designer to explicitly make moves in a game-like conversation with the problem [198]. It is so important that some researchers declare it to be the fundamental language of design, like mathematics is the language for scientific thinking [46]. The power of sketching, according to Cross, is contained in its ability to describe a partial model of a proposed design or problem. Detail can be subordinated, allowing a designer to zoom-in, solve a problem, and then abstract it away when returning to a high-level view. This has implications for software tools: designers must easily navigate the design space with undo, copy and paste, and a history of progress, creating tools with a "high ceiling" and "wide walls" [178]. We use the term "sketching" in a broad sense, including both pencil and paper and software or hardware sketches [27, 155]. Our design activity of *sketching* refers to exploration and demonstration as distinguished from iterative *refinement*. In other works, sketching can encompass iteration, annotation, and some level of refinement.

2.3.3 Collaboration

Design is a collaborative process with the potential for generating more varied ideas [227], and is important for creativity support tools [178, 205]. Although sometimes group dynamics influence the design process negatively, proper group management and sharing of multiple ideas results in more creativity and better designs [97]. Shneiderman in particular has championed collaboration in design [205], and suggests two different types of collaboration to be supported by creativity tools: relating, informal discussions with colleagues, and donating, disseminating information to the public/annals of time. Orthogonal to these intended purposes (relating and donating) is the collaboration context. Computer-supported collaborative work often separates interactions into four contexts ordered into two dimensions: collocated (same location) or distributed (different locations), and synchronous (simultaneous) or asynchronous (at different times) [66]. Collaboration is notable because it is inherently challenging to haptic design: two people can look at the same image or hear the same sound from across a room, but touch is a local sense, far easier in a collocated, synchronous setting. We explore informal collaboration briefly in Chapter 3, and explore the design activity of *sharing* in more detail in Chapters 6 and 7. Chapter 8 characterizes how professional haptic designers collaborate.

2.4 Previous Efforts for Haptic Experience Design

We are not the first to look into haptic media production or to apply design thinking to haptic design. While we present a cohesive look on HaXD process linked to general guidelines for HaXD support tools, previous work has developed authoring tools, supportive software and hardware platforms, and conceptual frameworks to facilitate HaXD.

2.4.1 Editors and design tools

As long as designers have considered haptic effects for entertainment media, they have needed compositional tools [88]. Custom editors (such as D-Box Motion Code Editor) and software plugins are provided to media designers that overlaid the visual and audio content with haptics, and allow designers to generate, tune and save frame-by-frame haptic content in an allocated track for it to play simultaneously with the media content.

By tuning parameters of these effects, users could personalize haptic content, embed it in games and share effects with other users. Similar devices and authoring schemes are also developed for online social interactions using custom multi-actuator haptic devices [123, 169, 223]. A great deal of previous work has focused on how to prototype or author haptic phenomena using non-programming methods.

Many user-friendly interfaces help designers create haptic sensations, especially with vibrotactile devices. The Hapticon editor [67], Haptic Icon Prototyper [215], and posVibEditor [184] use graphical mathematical representations to edit either waveforms or profiles of dynamic parameters (torque, frequency) over time. The Vibrotactile Score [132] was shown to be generally preferable to programming in C and XML, but required familiarity with musical notation [130]. The Demonstration-Based Editor [99] allows control of frequency and intensity by moving graphical objects on a touchscreen. Similar to the SPIN lab's Haptic Instrument (mHIVE, Chapter 3), this mobile tool was shown to be intuitive and easy to use for exploration or communication, but faltered when refining more elaborate sensations.

The Haptic Application Meta Language (HAML) [61] is an XML-based data format for adding haptics to MPEG-7 video, eventually augmented with the HAML Authoring Tool (HAMLAT) [62]. Abdur Rahman et al. [1] adapted an XML approach to YouTube, and Gao et al. [78] developed related online MPEG-V haptic editing. Augmented media experiences and HAV content [56], have used different methods of input. One approach is to use camera motion sourced from accelerometers [52] to actuate audience members' hands and head in a HapSeat [53, 55]. Later editable with H-Studio [54], this work has proposed the concept of Haptic Cinemotography [57], including basic principles of composition when combined with video [86]. Other approaches include automatic conversion of audio content Several studies have looked into automatic conversion from audio streams [36, 101, 131] or video streams [122] to VT or force-feedback output.

Commercially, Apple's end-user vibration editor has been present in iOS since 2011 (iOS 5) but only produces binary on/off timing information. Immersion provides two tools: TouchSense Engage is a software solution for developers, while Touch Effects Studio lets users enhance a video from a library of tactile icons supplied on a mobile platform. Vivitouch Studio allows for haptic prototyping of different effects alongside video (screen captures from video games) and audio, and supports features like A/B testing [216].

The control of multi-actuator outputs has been explored by TactiPEd [169], Cuartielles' proposed editor [50], and the tactile movie editor [123]; the latter combined spatial and temporal control using a tactile video metaphor for dense, regular arrays of tactile pixels ("taxels"), including a feature of sketching a path on video frames. However, these approaches embrace the separate control of different actuators, rather than a single perceived sensation produced by the multi-actuator device, which we address with tactile illusions in Chapter 4.

2.4.2 Platforms

There are many software libraries aim to support developers. The UPenn Texture Toolkit contains 100 texture models created from recorded data, rendered through VT actuators and impedance-type force feedback devices [51]. The HapticTouch Toolkit [128] and Feel Effect library [108] control sensations using semantic parameters, like "softness" or "heartbeat intensity" respectively. Vibrotactile libraries like Immersion's Haptic SDK (immersion.com) connect to mobile applications, augmenting Android's native vibration library. Force feedback devices have software platforms like CHAI3D (chai3d.org), H3D (h3dapi.org), and OpenHaptics (geomagic.com).

Hardware prototyping platforms like Arduino (arduino.cc) provide an open source microcontroller and development platform for physical prototyping. Phidgets (phidgets.com) facilitate rapid hardware prototyping with over 20 programming languages [83]. More recently, Wooden Haptics gives open-source access to fast laser cutting techniques for force feedback development [74], and faBrickation streamlines prototyping for 3D printing [158]. These platforms, especially Arduino, have made significant improvements to enable rapid iteration and hardware sketching. However, I believe we can do much better: these platforms require programming, hardware, and haptics expertise, and include inherent time costs like compilation, uploading, and debugging.

2.4.3 Conceptual Tools

Some higher-level perspectives offer outcome targets or design attitudes to guide haptic practitioners. "DIY Haptics" categorize feedback styles and design principles [95, 141]. "Ambience" is proposed as one target for a haptic experience [143]. Haptic illusions can serve as concise ways to explore the sense of touch, explain concepts to novices and inspire interfaces [93]. "Simple Haptics", epitomized by *haptic sketching*, emphasizes rapid, hands-on exploration of a creative space [155, 156]. Haptic Cinematography [57] uses a film-making lens, discussing physical effects using cinematographic concepts. The notion of distributed cognition [100] has particular relevance for haptic design, suggesting that people situate their thinking both in their bodies and in the environment. Haptics courses are taught with a variety of foci including perception, control, and design, providing students with an initial repertoire of skills [114, 167].

Haptics has often made use of metaphors from other fields. Haptic icons [140], tactons [17], and haptic phonemes [68] are small, compositional, iconic representations of haptic ideas. Touch TV [153], tactile movies [123], haptic broadcasting [34], and Feel Effects [108] attempt to add haptics to existing media types, especially video.

Musical analogies have frequently been used to inspire haptic design tools, especially VT sensations. The Vibrotactile Score, a graphical editing tool representing vibration patterns as musical notes, is a major example [130, 132]. Other musical metaphors include the use of rhythm, often represented by musical notes and rests [20, 23, 35, 221]. Earcons and tactons are represented with musical notes [17, 19], complete with tactile analogues of crescendos and sforzandos [22]. The concept of a VT concert found relevant tactile analogues to musical pitch, rhythm, and timbre for artistic purposes [88]. Correspondingly, tactile dimensions have been also been used to describe musical ideas [63].

The language of tactile perception, especially affective (emotional) terms, is another way of framing haptic design. Many psychophysical studies have been conducted to determine the main tactile dimensions with both synthetic haptics and real-world materials [67, 164]. Language is a promising way of capturing user experience [162], and can reveal useful parameters, e.g., how pressure influences affect [242]. Tools for customization by end-users, rather than expert designers, are another way to understand perceptual dimensions [200, 201]. However, this work is far from complete; touch is difficult to describe, and some even question the existence of a tactile language [111].

2.5 Methodology

We use mixed methods to explore our research questions of process, strategies, and experience. We begin by using qualitative techniques to gather rich, generative data from design tools and design processes, and to inform iteration. Through our work, we increasingly complement this data with quantitative methods, moving towards large-scale data collection for our generated theories with deployed tools.

In this dissertation, we draw from the philosophical and methodological traditions of phenomenology, and the methodology of grounded theory. Phenomenology is both a philosophical tradition and a social science methodology based upon that tradition that involves the study of subjective experience. We use Moustakas [157] as our primary guide through both, as it focuses on practical methodological concerns but provides a strong philosophical background; Creswell [45] provided an overview of various methodologies and resources for phenomenology. Critical components include *horizonalization*, or preparing oneself to consider all of the participant's statements equally and with fresh eyes; distinguishing and synthesizing *textural* descriptions, *e.g.*the participant's verbatim explanation of the experience, with *structural* description, or the analytical interpretation through psychology (or HCI) theories; and the documentation of the researcher's own experience with the phenomenon of study. Phenomology as a methodology has been used in psychology to investigate topics ranging from visual illusions to tactile experience [45, 162, 179].

Methodologies, like phenomenology, often include a set of methods combined with their philosophical and epistemological underpinnings. In this work, we specifically use the Stevick-Colaizzi-Keen methods as described by Moustakas [157]. Transcripts are divided into non-overlapping, non-redundant statements about the phenomena known as Meaning Units (MUs). This considers every statement that the participants make, and does not discount any due to bias or selective searching. Then, MUs are clustered into emergent themes through affinity diagrams, writing and re-writing of thematic descriptions, and reflection guided by phenomenological philosophy. We use this technique exactly in our first exploratory study (Chapter 3), and later combine it with Grounded Theory methods.

Grounded Theory is another well-known methodology first described by Glaser and Strauss [81]. We adopt the more flexible methodology described by Corbin and Strauss [44], as it allowed us to integrate with our phenomenological methods. We principally adapt the methods used in Grounded Theory, specifically, memoing (writing about each focused quotation or MU), constant comparison (comparing each new memo and codes to previous ones), and open and axial coding (creating codes, or concepts, linking them together, then categorizing statements or observations based on codes). This technique especially facillitated video analysis in

Chapter 4 and Chapter 5, and allowed for quantitative count-based data and simple statistics to complement our interview-based findings.

We also note that researchers who use methods from the umbrella term "qualitative research" often blend techniques from many methodologies. Phenomenology and grounded theory are two such methodologies, others include hermeneutics, ethnography [157], ethnomethodology, and thematic analysis [183]. For example, ethnography introduces the concept of "thick description" [], where the research tries to use detailed, evocative language to convey a rich sense of being in the observed environment. This technique is used more generally in observational notes and when writing up qualitative results to provide the rich data studied by qualitative methods.

While some design scholars adopt qualitative techniques [47, 48, 198], others have developed quantitative techniques. When studying graphic design for ads, Dow et al. [60] used ratings by experts as well as click-through rates and other online analytics for actual deployed ads from their study. Kulkarni et al. [125] used MTurk to generate sketches of aliens in several conditions (of exposure to examples), then deployed another MTurk task to label each drawing with features like antennae or feet. Lee et al. [129] had end-users rate graphic designs in both an in-lab study and over Mechanical Turk, and recorded time participant designers spent on each component. These approaches allow for hypothesis testing for specific research questions, but require infrastructure unavailable to haptics, notably, crowd deployment, rating scales, and mature input tools. In our early work, we found qualitative feedback to be sufficient while developing our understanding of how to build design tools. Later, we begin to approach this infrastructure, using an online editor with analytic logs in Macaron (Chapter 5), and examining the potential for crowdsourced feedback with HapTurk (Chapter 6). While a valuable goal, large-scale quantitative feedback on HaXD remains outside the scope of this dissertation.

Chapter 3

Sketch: The Haptic Instrument

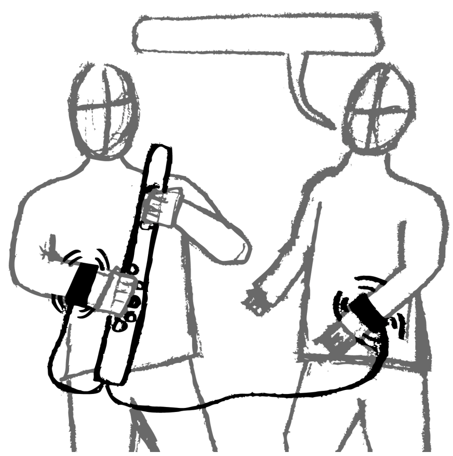


Figure 3.1: Concept sketch of a haptic instrument. Both users experience the same sensation, controlled in real-time.

Preface – The haptic instrument case study[1] was the first of our three vibrotactile design tools. We studied the role of real-time feedback and informal, synchronous collaboration on HaXD using musical instruments as inspiration. We recruited participants with haptics experience to serve as proxies for haptic designers. We built a haptic instrument, mHIVE, in a tablet-based interface, and used phenomenology to begin developing our evaluation methods. This study was small but seminal: we found mHIVE was effective for exploration but not refinement, which lead to distinguishing the design activities of *sketch* and *refine*.

[1]Schneider and MacLean. (2014) *Improvising Design with a Haptic Instrument*. HAPTICS '14.

3.1 Abstract

As the need to deploy informative, expressive haptic phenomena in consumer devices gains momentum, the inadequacy of current design tools is becoming more critically obstructive. Current tools do not support collaboration or serendipitous exploration. Collaboration is critical, but direct means of sharing haptic sensations are limited, and the absence of unifying conceptual models for working with haptic sensations further restricts communication between designers and stakeholders. This is especially troublesome for pleasurable, affectively targeted interactions that rely on subjective user experience. In this paper, we introduce an alternative design approach inspired by musical instruments – a new tool for real-time, collaborative manipulation of haptic sensations; and describe a first example, mHIVE, a mobile Haptic Instrument for Vibrotactile Exploration. Our qualitative study shows that mHIVE supports exploration and communication but requires additional visualization and recording capabilities for tweaking designs, and expands previous work on haptic language.

3.2 Introduction

Haptic feedback has hit the mainstream, present in smartphones, gaming and automobile design, but our knowledge of how to design haptic phenomena remains limited. There are still no agreed-upon vocabularies or conceptual models for haptic phenomena [67, 128, 132, 162], in contrast to other modalities (*e.g.*, using theory of minor chords to evoke a sad emotion in music). For subjective qualities, such as pleasant alerts or frightening game environments, prospects are even more limited. Design is still based on trial and error with programming languages, limiting exploration. The lack of established conceptual models or design frameworks further challenges communication between designers and stakeholders.

Using a music composition metaphor (as in [132]), we are writing music without ever playing a note. Instead, we compose a work in its entirety, then listen to the result before making changes. In contrast, musicians often use their instruments as a tool for serendipitous exploration when designing music and can draw upon musical theory. Furthermore, music is collaborative, with communication facilitated by a reference point of a sound. Touch, however, is a personal, local sense, making it difficult to discuss stimuli.

Facilitated exploration and collaboration should streamline the haptic design process and inform a guiding theory, analogous to those for musical composition. Designers will attain fluency with new devices and control parameters, while collaborative elements will get people designing in groups. A usable haptic language may emerge from their dialogue.

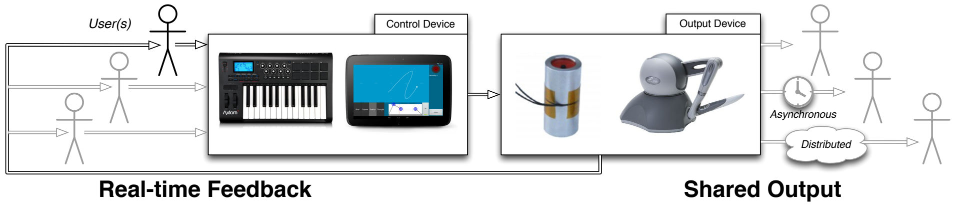


Figure 3.2: The haptic instrument concept. One or more people control the instrument, and receive real-time feedback from the device. Any number of audience members can feel the output in real time as well. Control methods can vary, from traditional musical control devices (such as the M-Audio Axiom 25, used in preliminary prototypes) to touchscreen tablets (used in mHIVE). Output devices vary as well.

Our approach is to directly address these shortcomings with the development of a *haptic instrument*, inspired by musical instruments but producing (for example) vibrotactile sensations rather than sound (Figure 3.1). Haptic instruments have two main criteria: they provide real-time feedback to the user to facilitate improvisation and exploration, and produce haptic output to multiple users as a *what-you-feel-is-what-I-feel* (WYFIWIF) interface. This allows for a dialogue that includes a haptic modality: haptic instruments create a shared experience of touch, allowing for a common reference point. We developed a vibrotactile instance, mHIVE (mobile Haptic Instrument for Vibrotactile Exploration), as a platform to investigate this concept. Our main contributions are:

- A definition of the haptic instrument concept & design space.
- A fully-working haptic instrument (mHIVE).
- The novel application of an established psychological methodology, phenomenology, to investigate mHIVE's interface and subjective tactile experiences.
- Preliminary results from a qualitative study that show mHIVE supports exploration and collaboration, and implications for the design of future haptic design tools.

then define the haptic instrument, its requirements, features, and design space. We report the design of mHIVE, our methodology, and preliminary results, and conclude with future directions for haptic tool design and research into a haptic language.

3.3 Related Work

We cover previous work related to musical metaphors for haptic design, other tools for haptic design, and the language of haptics.

3.3.1 Musical Metaphors in Haptic Design

Musical analogies have frequently been used to inspire haptic design tools. The vibrotactile score, a graphical editing tool representing vibration patterns as musical notes, is an example of controlling vibrotactile (VT) sensations [130, 132]. The vibrotactile score provides an abstraction beyond low-level parameters and can draw from a musician's familiarity with the notation, but we can take this idea further: when writing a song, a musician might improvise with a piano to try out ideas. We are inspired by the vibrotactile score and musical instruments, but define haptic instruments as a more general concept than literal musical instruments for touch.

Other musical metaphors include the use of rhythm, often represented by musical notes and rests [20, 23, 35, 221]. Tactile analogues of crescendos and sforzandos have proven valuable to designing changes in amplitude [22]. Indeed, Brewster's original earcons and tactons were represented with musical notes [17, 19]. The concept of a vibrotactile concert or performance was explored to identify relevant tactile analogues to musical pitch, rhythm, and timbre for artistic purposes [88]. As well, tactile dimensions have been used to describe or map to musical ideas [63]. Musical concepts have been widely used in the design of vibrotactile sensations, which we draw upon when designing mHIVE.

3.3.2 Other Haptic Design Approaches

Many tools have been developed to make it easier to work with the physical parameters of a haptic device. The Hapticon Editor is a graphical software tool that allows direct manipulation of the waveform for vibrations [67], and in another approach, piecing together of smaller iconic idioms [215]. This idea is best encapsulated by "haptic phonemes", the smallest unit of meaningful haptic sensations that can be combined [68]. A similar approach was used with TactiPEd, a graphical metaphor for control of wrist-based actuators, by controlling the low-level parameters of frequency, amplitude, and duration [169]. Haptic instrument parameters can be low- or high-level, but we use similar parameters with mHIVE.

Non-graphical approaches have also contributed to haptic design. Programming has benefitted from the use of toolkits such as HapticTouch, which uses higher-level descriptors ("Softness", "Breakiness") to control tangibles [128]. Though a promising direction, the vocabulary is not empirically grounded, and developers still have to deal with physical parameters. Hardware sketches and designing through making are also important approaches, since the immediate feedback of being able to feel haptics is crucial [156].

3.3.3 Haptic Language

Investigation into the language of tactile stimuli has a long history in psychological studies [164]. Many psychophysical studies have been conducted using factor analysis or similar approaches to determine the main tactile dimensions [164], but these have looked at materials rather than synthesized vibrotactile sensations, and have primarily been deductive (evaluating a pre-determined set of terms) rather than inductive (asking participants to describe sensations without prompting). Other work has shown little consensus on constant meanings for difference tactile dimensions, or whether a tactile language even exists [111]. There is a clear need to empirically investigate the subjective experience of touch-based interfaces, for which phenomenology is ideal [45, 157].

Our study is perhaps most closely related to Obrist, Seah, and Subramanian's work on the perception of ultrasound transducers [162]. Their study examined the language used to describe two different sensations, one oscillating at 16 Hz and the other at 250 Hz. Though they also used phenomenology, our study differs in two important ways: we explore vibrotactile sensations rather than ultrasound, and give our participants a way of controlling the phenomenon directly, allowing for more coverage of the stimulus design parameters. A more deductive approach by Zheng and Morell also looked at how pressure and vibration actuators influenced affect, noting that affect influences attention, and documented qualitative descriptions of the sensations [242].

3.4 Defining the Haptic Instrument

We define a haptic instrument as a tool for general manipulation of one or more haptic (tactile, force-feedback, or both) devices that provides real-time feedback to anyone controlling the device, and can produce identical shared (WYFIWIF) output to all users to facilitate discussion and collaboration. Manipulation can include ideation, exploration, communication, recording, refinement, and articulation. Manipulation can be for utilitarian purposes (*e.g.*, designing haptic notifications) or artistic expression (*e.g.*, a haptic performance). Output devices can be purely output, or interactive. Furthermore, although haptic devices must be involved, multimodal experiences could easily be created by combining a haptic instrument with auditory or visual output.[2]

[2]One could even imagine a multimodal instrument such as Asimov's Visi-Sonor [6] or its parody, Futurama's Holophonor [75].

3.4.1 Design Dimensions

There are several main design dimensions that can be considered in a haptic instrument (outlined in Figure 3.2). A haptic instrument can occupy multiple positions on these dimensions.

Asychronous/synchronous. Though a haptic instrument must provide real-time feedback, its collaborative (shared-output) aspect could be either synchronous (by having multiple people experience the real-time output) or asynchronous (by allowing for recording and playback, important for design).

Collocated/distributed. A haptic instrument's output could be present only for users in the same room, or be broadcast over a network to people around the world. For example, multiple mobile devices could all display identical output in a distributed manner.

Private/shared control. A haptic instrument's control could be private (operated by a one person at a time) or shared (multiple users control the display). Shared control could be collocated or distributed (*e.g.*, a web interface and shared object model).

Output mechanism. Each haptic instrument will control a haptic device, which has its own mechanism for providing a haptic sensation (*e.g.*, vibrotactile sensations). Because haptic devices can be complex and combine multiple mechanisms, this is a large space in its own right. Characterizing the different display mechanisms is something that we must leave to future work. Suffice it to say, a haptic instrument will be different depending on its output device.

Number of haptic instruments or output devices. One consideration is whether a haptic instrument is intended to operate alone, or with other haptic/-multimodal instruments. One can imagine haptic jam sessions for inspiration and ideation, or even form haptic bands for artistic expression. This is highly related to private/shared control – there is a fine line between several identical haptic instruments with private control, and a single haptic instrument with shared control and several output devices. Note that a haptic instrument may involve several devices to produce shared-output.

Control mechanism. Similarly, a haptic instrument could be controlled in a variety of ways. From musically-inspired MIDI controllers to smartphone applications, we envision a wide variety of control methods. Even a real-time programming environment might be appropriate for complex interactive sensations, so long as the control mechanism works with the output device's paradigm.

We expect that haptic instruments could provide both immediate and long-term value. We hope haptic instruments will improve the design process immediately, by supporting exploration and collaboration. Over time, their use could lead to a natural, emergent design language valuable in its own right. One can also imagine a general tool composed of several virtual haptic instruments, much like digital musical synthesizers.

3.5 mHIVE

We developed mHIVE to begin to explore how a haptic instrument should work and what it should do (Figure 3.3). mHIVE is collocated, synchronous, and mostly private in control; it accommodates shared display via dual Haptuators [233] and is operated with a single-touch tablet-based interface (**??**). We began with vibrotactile design because VT sensations are common, do not require interactive programming, are controlled through waveforms (analogous to music), and their low-level control parameters are well understood. A touchscreen allowed direct manual control.

mHIVE offers real-time control of frequency, amplitude, waveform, envelope, duration, and rhythm, identified as the most important parameters for vibrotactile sensations [17, 22, 23, 88, 180].

The main view controls amplitude (0 to 1) on the vertical axis, and frequency (0-180Hz, determined by piloting) horizontally. Amplitude and frequency were combined because we modeled them both as continuous controls: dynamics of continuous amplitude have been shown to be a salient design dimension [22, 88], and we did not want to choose discrete bins for frequency at this early stage. Further, single-handed control was essential – the other hand is required to feel the output. These axes were labeled to help users understand what they were and to give general sense of the values. A two-dimensional visual trace shows the previous two seconds of interaction history with the main view, intended to provide feedback and aid memory about drawings that were used.

VT duration and rhythm are directly mapped to screen-touch duration and rhythm. In analogy to musical timbre [17, 88], we provided four waveforms: sine, square, rising sawtooth and triangle. Sine and square are distinguishable [88], but we added sawtooth and triangle waveforms to expand the palette.

The attack-decay-sustain-release (ADSR) envelope controls amplitude automatically as duration of the note continues, as a 0-to-1 multiplier of the amplitude displayed on the main amplitude-frequency input. Attack determines the amount of time (in milliseconds) to ramp the amplitude from 0 (none) to 1 (full). Decay determines the amount of time (in milliseconds) to ramp the amplitude from 1 (full) to the sustain level. Sustain determines the amplitude level (from 0 to 1) held as long as the user keeps a finger on the display, playing a haptic note. Release determines the amount of time (in milliseconds) to ramp the amplitude from the sustain level to 0 (none). This envelope is a common feature of synthesized or digital musical instruments, and was noted as particularly useful in the Cutaneous Grooves project

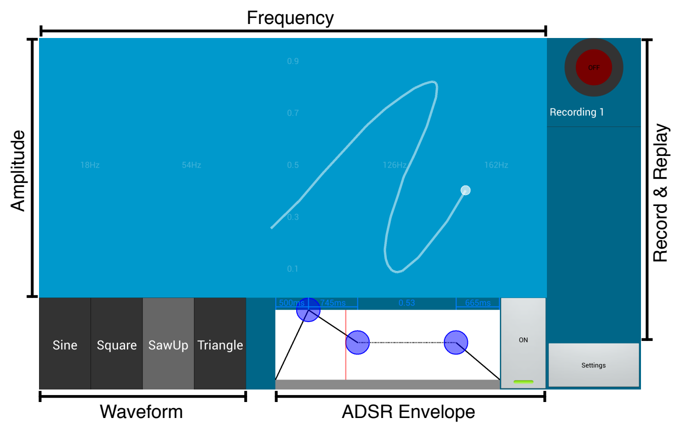


Figure 3.3: mHIVE interface. Primary interaction is through the amplitude-frequency view, where visual feedback is provided through a circle (current finger position) and a trail (interaction history).

[88].

During piloting, we noticed that the ADSR concept was difficult to explain. We thus developed a novel interactive visualization, where the user could change the envelope parameters by dragging circles around. A red line operates as a cursor or playhead, showing the current progress through the envelope, looping around the dotted line when the sustain level is held.

Recording functionality was added to support more advanced rhythms and repetitions, and to allow users to save their sensations for later comparison. The record feature captures changes in frequency, amplitude, waveform, ADSR, and *replayed recordings*, allowing for compound haptic icons to be created. During playback, all changes are represented in the interface as if the user had manipulated them in real-time. At this time mHIVE only produces a single output sensation (with a single waveform, ADSR setting, frequency, and amplitude). Multitouch, layering, and sequencing (automatically playing multiple notes with a single touch) are not supported, as the semantics were too complex for a first design.

mHIVE is implemented in Java using the Android SDK [4], and the FMOD sound synthesis library [73] to produce sounds, sent to two or more Haptuators through an audio jack. We deployed mHIVE on an Android Nexus 10 tablet running Android 4.2.1.

3.6 Preliminary Study Methodology

We conducted a preliminary qualitative study to investigate two questions. First, is mHIVE an effective tool for the expression, exploration, and communication of affective phenomena? Second, what language, mental models, and metaphors do people use to describe vibrotactile sensations, and how do they relate to mHIVE's low-level control parameters?

We collected and analyzed our data using the methodology of phenomenology, an established variant of qualitative inquiry used in psychology to investigate topics ranging from visual illusions to tactile experience [45, 162, 179]. Phenomenology explores subjective experience, appropriate for an investigation into the more intangible qualities of pleasantness and affect. At this point, the rich, inductive data of qualitative analysis is more valuable than a controlled experiment with statistical analysis.

In particular, we use the Stevick-Colaizzi-Keen method as described by Moustakas [157]. In-depth interviews are conducted with a small number of participants. The interviewer, Researcher 1 (R1), also documents his experience, as if he was interviewing himself. Then, R1 transcribes each interview, including his own. Transcripts are divided into non-overlapping, non-redundant statements about the phenomena known as Meaning Units (MUs). This considers every statement that the participants make, and does not discount any due to bias or selective searching. Then, MUs are clustered into emergent themes. We interpret our themes in the Discussion.

3.6.1 Procedure

Our 1-hour open-ended interviews used the following protocol:

1. Ask the participant for their background: occupation, experience with touch-screens, haptics, music, and video games.

2. Demonstrate mHIVE to the user, and invite them to explore while thinking aloud to describe the sensations they feel.

3. Probe the design space by asking participants to explore different control parameters, and to explore their metaphors (*e.g.*, if the participant describes a sensation as "smooth", R1 would ask them to try to produce a "rough" sensation).

4. Ask the participants to produce sensations for the six basic cross-cultural emotions documented by Ekman [65], and rank how well they think their sensation represents the emotion on a 4-point semantic differential scale

Figure 3.4: Study setup. Both the participant (left) and the interviewer (right) feel the same sensation as the participant controls mHIVE.

(Very Poorly, Somewhat Poorly, Somewhat Well, Well). This was done both as an elicitation device to gather a wider range of interactions with mHIVE, and to directly investigate a design task.

5. Set the Haptuators down, and ask the participants to describe their experience of working with mHIVE in as complete detail as possible to evaluate the device itself.

R1 conducted the interviews and analysis, which required specialized knowledge of mHIVE. Scores of inter-rater reliability common with other qualitative analyses (*e.g.*, grounded theory [44]) are inappropriate and unavailable, as we did not conduct deductive, low-level coding. To improve reliability, R1's documented experience was analyzed first, and then consulted during analysis to remove bias (*e.g.*, to not use terms only used by the experimenter).

3.7 Results

We sought participants with experience designing haptics as a proxy for expert designers for our initial study. Four participants were recruited through email lists and word-of-mouth (P1-4, three male), and were all in the age range of 26-35 with self-reported occupations including graduate students or post-docs in information visualization, HCI, and human-robot interaction). All had experience working with haptic technology, and (because of this requirement) all knew the main researcher in a professional capacity, although only P2 had seen earlier prototypes of the haptic instrument. The small sample size, typical for phenomenological studies [45], was appropriate for the rich data we wanted. Data collection ended when we achieved saturation of new results, and had a clear direction for our next iteration.

Here we report the three major themes that emerged during analysis: mHIVE's success as a haptic instrument, mHIVE's limitations that reveal more detail about the haptic design process, and the use of language in the study.

3.7.1 mHIVE Succeeds as a Haptic Instrument

Our results suggest that mHIVE can be effective for exploration of a design space, and communication in the haptic domain. Overall, mHIVE was well received, seen as a novel and promising tool. *"I definitely liked it"* (P1), *"I think there should be more devices like this for designing haptic icons"* (P2).

Serendipitous exploration. Participants reported that mHIVE was best served to explore the design space, generate a number of ideas, and try things out. Serendipitous discoveries and exclamations of surprise were common. Participants were able to *"accidentally stumble upon something"* (P2) as they explored the device. *"I felt I could get a large variety"*, *"I could easily play around with the high-level to find out what was neat"* (P3).

Communication. mHIVE established an additional modality for dialogue. The dual outputs created a shared context, demonstrated by deictic phrases: the additional context of the vibrotactile sensation was required to make sense of the statement. The use of "that" and "there", reminiscent of the classic "Put That There" multimodal interaction demo [14] indicate a shared reference point was established from the haptic instrument. *"So there'd be like, (creates a sensation on the device), which is pretty mellow"* (P3).

In particular, P4 successfully communicated the sensation of sleepiness to the R1, by asking whether R1 could guess the sensation. *"Can you guess it?"* (P4) *"Sleepy?"* (R1) *"Yeah. Pretty good"* (P4). The dialogue worked as a two way channel, as R1 was able to phrase questions using the device. *"It was different"* (P2) *"How was it different?"* (R1) *"You delayed the first part, it felt new"* (P2).

Certain sensations, like a feeling of randomness, could only be felt when another person controlled mHIVE. *"When someone else does it, I feel better, it's like, you cannot tickle yourself"* (P2).

3.7.2 Tweaking through Visualization and Modification

During analysis, some key directions for future design emerged around visualization and control capabilities.

Inability to tweak. Though mHIVE supported exploration and collaboration, we found it was inadequate as a standalone design tool. Few created sensations were considered to be final. Many descriptions were hedged and in the design task, few sensations captured the emotional content well. *"I dunno, maybe that's afraid?"* (P1), *"Still felt that you can make them better"* (P2), *"To me that's more* *fuming (laughing) than it is angry"* (P3). On some occasions, participants were certain about their descriptions. *"Sad, definitely down on the amplitude with sad... oh that's totally sad. Yeah."* (P1). This was uncommon, and usually tied to discovering an ideal sensation during the design task.

More visualization and recording. Part of mHIVE's inability to support tweaking was due to cognitive limitations for both memory and attention. Participants found it difficult to remember what they had tried before, and to pay attention to the output while simultaneously controlling it. *"There's a lot of variables which, when I'm trying to compare between two configurations... it was hard sometimes to remember what I had tried"* (P3), *"I definitely liked being able to feel a stimulus without having to implement it, you know, it allows me to focus more on what it feels like"* (P1).

Participants suggested that although visualization and recording features helped somewhat to overcome these limitations, more was needed. All requested greater emphasis on recording through repetition or looping, both to aid memory and allow for focus on the sensation independent of device control.

Allowing persistent, modifiable sensations and alternative visualizations could also help participants overcome these limitations. *"The recording records what I do, but it'd be nice to have it repeat stuff"* (P3), *"It might conceivably be nice to be able to, you know, draw a curve, draw a pattern, draw like you would in paint, and then be able to manipulate it, replay it, move the points, see what happens"* (P1).

3.7.3 A Difficult Language

Our study was too small to analyze language patterns in detail, but exposes emerging trends.

Pleasantness, ADSR, and frequency. Participants often started with a statement of like or dislike rather than a description. Pleasant sensations often involved the ramp-in and ramp-out ("echo" or "ringing") of the ADSR envelope, or lower-frequency sensations. Longer, higher frequency without ramp-in and ramp-out were less pleasant. *"I don't know how else to describe it, I kinda like it"* (P1), *"Yeah, this [ADSR] seems natural, somehow"*, *"It feels unnatural to kill the echo right away"* (P2), *"I like this [low-frequency] sensation cuz to me it feels a lot like purring"* (P3).

Waveform. Participants all noticed differences between waveforms, but were often challenged in expressing them (P4 used the musical term "timbre"). Square waves in particular were distinct, with a greater range and stronger affinity to mechanical sensations. *"It's interesting, they feel more different than I thought they would"* (P1), *"If you want to make something feel like a motorcycle, you would definitely need square wave"* (P2).

Aural/haptic metaphors drawn from previous experience. For the most part, participants used concrete examples and direct analogies to describe sensations, often drawn from their previous experiences. One stand-out strategy employed by all participants was onomatopoeias: *"beeooo"* (P1&4), *"vroom"* (P1), *"bsheeeooo"*, *"boom"*, *"neeeaa"*, *"mmmMMMmmm"* (P2), *"pa pa pa pa"*, *"tum tum tum tum"*, *"tumba tumba tumba tumba"* (P3); *"upward arpeggio, like, (singing with hand gestures) na na na naaa"* (P4). Other sound-based metaphors were very common, including hum, buzz, whistle, rumble (P1); bell (P1, P2); squeaky, creak (P2); or thumpy (P3). Still other descriptors were directly haptic in nature: rough, flat (P1); sharp, round, ticklish (P2); sharp, smooth, cat pawing (P3); impatient foot tapping (P4).

3.8 Discussion

Here we interpret these themes to draw implications for haptic design tools, and compare to research on the language of haptics. We then reflect upon our methodology and limitations.

3.8.1 Design Tools

mHIVE was able to achieve the two main goals of a haptic instrument, facilitating both exploration and collaboration. Participants were clearly able to explore the different low-level parameters, and encountered serendipitous or unexpected sensations through improvisation. mHIVE created a shared experience that facilitated communication between R1 and the participants. We can thus conclude that haptic instruments are a promising new tool in a haptic designer's arsenal, with a first, successful implementation in mHIVE.

However, the second theme shows that serendipity and communication are only part of the equation. mHIVE does not serve as a general editor of haptic sensations. In particular, participants found their attention split when controlling the device and feeling the sensation; perhaps the real-time control should allow for a rapid, but not instantaneous, switch in focus between control and perception. More generally, participants were unable to tweak sensations because there was insufficient support for comparing ideas or evolving an existing idea.

In hindsight, this general difficulty is understandable given the broader context of the musical instrument analogy we used for inspiration. Musical instruments are not used to write songs on their own, but combined with notation or recording media. A similar combination of a haptic instrument and recording might be described more succinctly as a *haptic sketchpad*. Sketching is critical in design because it allows for the evolution of an idea through multiple sketches, as well as criticisms, comparisons, and modifications [48]. Emphasizing a history feature that supports multiple versions of sketches, the user could develop an idea as if with a multiple pages in a sketchbook. Haptic sketching in hardware has already been shown to be effective [156]. As well, a visual metaphor resonates with the desire for more effective visualization.

Ultimately, haptic instruments may be most useful as one element in a suite, or component of a more general tool. A haptic instrument could complement a graphical editing tool that does support tweaking, such as the vibrotactile score [130, 132] or the hapticon editor [67]. As part of a more comprehensive tool, mHIVE could be improved to reduce cognitive barriers to memory and attention. Alternatively, we could add functionality to mHIVE to support looping, visualization, and direct manipulation of the sensations within the tool. We will explore these options as we iterate on mHIVE's design in future work.

3.8.2 Language

Our preliminary results for language are compatible with the literature, supporting previous work. Participants' readiness to say whether a sensation was pleasant or not supports the view that touch is affective in nature, and that knowing what one likes or doesn't like is a primary function of touch [111]. ADSR pleasantness and high-frequency unpleasantness are both consistent with the literature: Zheng and Morell note that ramped signals influenced affect more positively than step signals, and 3s high-frequency sensations were annoying or agitating [242]. The heavy use of onomatopoeias is reminiscent of Watanabe *et al.*'s work with static materials [228]. However, in our study, onomatopoeias were often used to express dynamic sensations (beeeooo being a gradual decrease in amplitude and frequency), which might be a useful direction for future work.

3.8.3 Methodology and Limitations

Although phenomenology is uncommon in the haptics community (excluding [162]), we found it to be an effective way to empirically examine the subjective experience of using mHIVE. Because the community is still developing processes and tasks for haptic design, qualitative studies seem to be an especially appropriate way to tackle these problems. Once we have further defined haptic design, we can then move to more task-based, experimental methods.

Our study was a first round of feedback to inform our next iteration, and has limitations. First, our participant pool is (intentionally) small, and participants were all collected through our professional network, as people with haptic design experience are rare. As we continue to tackle the problem of haptic design, we hope to seek out a larger and more diverse pool of participants, and explore more realistic design tasks.

3.9 Conclusion

In this paper, we have introduced the concept of the haptic instrument, a new tool for haptic designers that supports serendipitous exploration and collaboration. We described the implementation of mHIVE, a mobile Haptic Instrument for Vibrotactile Exploration, with design decisions drawn from the literature. Our findings suggest that haptic instruments are effective tools for improvised exploration and collaboration, but only support part of the design process. Additional tools or features are required to support tweaking. Finally, we reported the use of language when interacting with mHIVE, expanding upon several conclusions in the literature.

We believe this to be a step towards a greater goal, the establishment of haptic design as its own discipline, with processes, tools, and best practices. Future work will build on this base as we continue to examine the haptic design process. We will consider a haptic sketchpad concept as one way to overcome the cognitive barriers, and allow users to tweak their designs. We also hope to apply haptic instruments and other tools in more realistic design scenarios. By supporting designers at this critical point, we can continue to make haptics more valuable than ever.

3.10 Acknowledgments

This work was supported by NSERC and approved by UBC's Behavioural Research Ethics Board (#H13-01620). We are very grateful to our participants.

Chapter 4

Refine: Tactile Animation

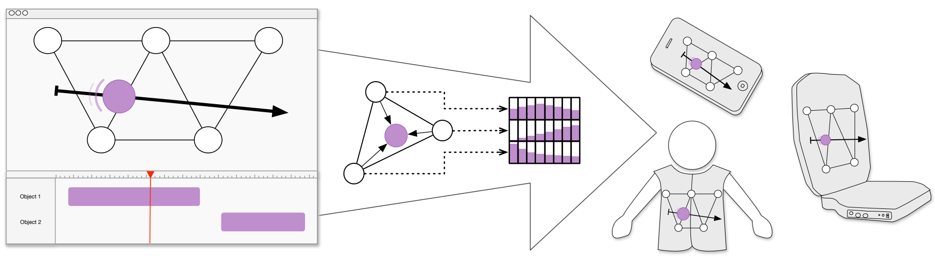


Figure 4.1: Concept sketch for tactile animation. An artist draws an animated sequence in the user interface and the user experiences phantom 2D sensations in-between discrete actuator grids.

Preface – In this second case study[1], we iterate on our findings from the haptic instrument to build a full authoring tool that supported both *sketching* and *refinement*. This work expanded to spatial vibrotactile designs with professional non-haptic media designers. We surveyed critical haptic authoring tool features and developed a full rendering pipeline for the tactile animation object, an abstraction able to handle diverse spatial vibrotactile arrays. We evaluated the implemented tool, Mango, with both phenomenology and methods from grounded theory and iterated on our study tasks. Professional animators transferred their non-haptic design skills to both explore (sketch) and iterate (refine), but missed features to reuse design elements and gather inspiration from examples. This theme, also glimpsed in Chapter 3, lead to the third design activity: *browse*.

[1]Schneider, Israr, and MacLean. (2015) *Tactile Animation by Direct Manipulation of Grid Displays*. UIST'15.

4.1 Abstract

Chairs, wearables, and handhelds have become popular sites for spatial tactile display. Visual animators, already expert in using time and space to portray motion, could readily transfer their skills to produce rich haptic sensations if given the right tools. We introduce the *tactile animation object*, a directly manipulated phantom tactile sensation. This abstraction has two key benefits: 1) efficient, creative, iterative control of spatiotemporal sensations, and 2) the potential to support a variety of tactile grids, including sparse displays. We present Mango, an editing tool for animators, including its rendering pipeline and perceptually-optimized interpolation algorithm for sparse vibrotactile grids. In our evaluation, professional animators found it easy to create a variety of vibrotactile patterns, with both experts and novices preferring the tactile animation object over controlling actuators individually.

4.2 Introduction

Haptic feedback is viewed today as a key ingredient of immersive media experiences. Body-moving devices in theatre seats, ride vehicles, and gaming platforms can tilt, translate, and shake the user for increased engagement. Recently, arrays of multiple actuators have been developed to display expressive, spatial sensations on the skin [53, 106, 123, 210, 231].

Vibrotactile (VT) arrays, which stimulate the skin through vibration, are common in diverse applications from immersive gaming chairs [106] to wearable vests for mobile awareness [115]. These displays typically employ sparse actuator arrangements to reduce cost and power requirements, using perceptual illusions to create continuous sensations [3, 105, 203]. Unfortunately, adoption of VT arrays is limited by a lack of authoring tools. Most only support a single actuator [67]; those that accommodate multiple actuators control each separately [123, 169, 216], cumbersome for non-adjacent actuators.

To remedy this, we propose the *tactile animation object*, an abstract, directly manipulable representation of a phantom sensation perceived in-between physical actuators. With this approach, designers can efficiently and creatively explore ideas and iterate without worrying about underlying actuator arrangements. As long as a rendering algorithm can be developed, this abstraction not only facilitates design, but is compatible with a variety of form factors and technologies.

In this paper, we describe the tactile animation object and implement it in *Mango*, a tactile animation tool and pipeline (Figure 4.1). Our contributions are: 1) A tactile animation interface grounded in user interviews and prior literature. 2) A rendering pipeline translating tactile animation objects to phantom sensations on sparse, generalized VT arrays, optimized with a perceptual study. 3) An evaluation with professional animators showing accessibility and expressivity. 4) An exploration of potential applications for tactile animation.

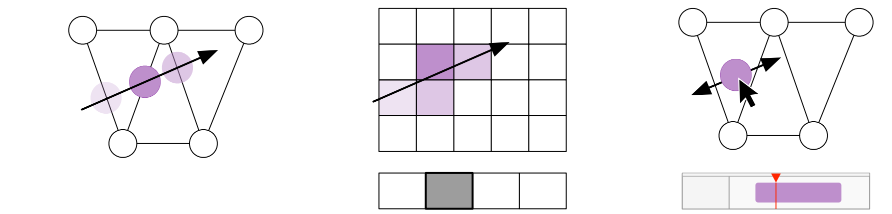
4.3 Background

4.3.1 Haptic Entertainment Technologies

Haptic feedback was used in cinema as early as *Percepto*, a 1959 multisensory experience for the movie "The Tingler" [103] with theater seats that buzzed the audience at strategic moments. Current 4D theaters, rides, shows, and gaming arcades are equipped with sophisticated motion platforms (e.g., D-Box, www.d-box.com) that supplement visual scenes. Large tactile transducers (such as Buttkickers, www.thebuttkicker.com) that shake the entire seat using the sound stream are also common with gaming and music content. Custom editors (such as D-Box Motion Code Editor) and software plugins overlay visual and audio content with haptics, and allow designers to generate, tune and save frame-by-frame haptics in an allocated track.

In contrast to displacing the entire body, multichannel haptic devices create percepts of dynamic and localized haptic sensations on the user's skin [106] and in mid-air [231]. Similar devices have been developed for online social interactions using custom multi-actuator displays [123, 169, 223]. All of these technologies require extensive programming experience, knowledge of hardware and background in haptic sciences to generate expressive and meaningful haptic content. Without guiding principles or haptic libraries, content generation schemes are complex, device-specific, and time consuming.

Another class of haptic technology renders high-resolution spatio-temporal patterns on the skin using a sparse array of VT actuators. These technologies use parametric models of sensory illusions in touch, such as phantom tactile sensations [3], and create illusory vibrations in between two or more VT actuators. This idea has been used to create a perceived motion flow between two vibrators mounted on the ends of a handheld device [203] and to create across-the-body and out-of-the-body illusions on a mobile device using up to four actuators [133]. The Tactile Brush algorithm [105] combined phantom tactile sensations and apparent tactile motion to render high-resolution and moving haptic patterns on the back using a coarse grid of VT actuators, but paths must be pre-determined (Figure 4.2a). Other spatio-temporal VT illusions such as the "cutaneous rabbit" [218] and Tau and Kappa effects [93] can be also used with VT arrays.



(a) Tactile Brush [105]: precomputed paths **(b)** Tactile Video [123]: frames of tactile pixels **(c)** Tactile Animation: direct manipulation

Figure 4.2: Comparison between related systems.

4.3.2 Haptic Authoring Tools

As long as designers have considered haptic effects for entertainment media, they have needed compositional tools [88]. Requirements drawn from previous work on how to prototype, sketch, or control haptic phenomena using non-programming methods are summarized in Table 4.1.

The Hapticon editor [67], Haptic Icon Prototyper [215], posVibEditor [184], and Immersion's Haptic Studio (www.immersion.com) use graphical representations to edit either waveforms or profiles of dynamic parameters (such as frequency or torque) over time. Another approach is predefining a library of haptic patterns to augment media content. Immersion Corporation's Touch Effects Studio lets users enhance a video from a library of tactile icons supplied on a mobile platform. Vivitouch Studio [216] allows for haptic prototyping of different effects alongside video (screen captures from video games) and audio. These tools focus on low-level control of device features rather than a semantic space, and control devices with either a spatial or temporal component, but not both simultaneously.

Several tools have allowed users to author haptic content using accessible touch-screen interactions. A demonstration-based editor [99] allowed control of frequency and intensity by moving graphical objects on a screen. mHIVE [191] controls frequency, intensity, waveform and envelope of two tactors with touch-screen gestures. Both systems were shown to be intuitive and easy to use for exploration or communication, but faltered when refining more elaborate sensations. Commercially, Apple's vibration editor (since iOS 5, 2011) allows users to create personalized vibratory patterns by touching the screen, but only produces binary on/off timing information.

Other aids to creating haptic phenomena include haptic sketching [156] for hands-on exploration of haptic ideas in early design, and end-user customization of tactile sensations [200]. Both emphasize exploration and broad manipulation

LR	Description
LR1	**Real-Time Playback** [156, 191] Rapid prototyping is essential for working with VT sensations, especially in absence of objective metrics. Feeling a sensation at design time allows iteration to converge faster to better results. However, *too* real-time can cause split attention.
LR2	**Load, save, manipulate** [112, 178, 191] A persistent object model is essential for sensation editing over longer projects and sharing with other designers or across devices. Well-defined actions upon a data structure also facilitates features like *undo* that support experimentation.
LR3	**Library of effects** [67, 97, 169, 215, 216] A library of saved sensations is an important feature used in previous haptic authoring tools, providing inspiration and preventing designers from re-inventing the wheel.
LR4	**Device configuration** [123, 130, 131, 169] Because of the many types of haptic devices, a general tool must be able to understand different devices. Lightweight configuration files are common in the literature, allowing users to select specific hardware, specify location and type of actuators, and choose a rendering algorithm.
LR5	**Multiple channels & combination of effects** [67, 169, 184, 215, 216] Being able to display multiple effects simultaneously, or combine effects via superposition or concatenation, is essential for expanding the design space. This is typically represented in a timeline, which represents the temporal behaviour of any objects.
LR6	**Visual/direct control metaphor** [50, 123, 169] Most previous tools consider each actuator separately. When thinking semantically about a spatial system, a direct view of the device and actuator layout is critical for direct manipulation.
LR7	**Audio/visual context** [123, 156, 216] Haptic perception depends greatly on additional senses [93]. By providing audio and visual feedback, these effects can be mitigated and the designer can experience haptic sensations in context.
LR8	**User Feedback** [191, 216] Receiving feedback from users, either by demonstration or A/B testing, is extremely valuable.

Table 4.1: Literature Requirements (LRs) for a tactile animation authoring.

rather than finely controlled end results. HAMLAT [62] supports authoring of force feedback in static 3D scenes. Lee and colleagues [130] used a musical metaphor for vibrotactile authoring. Schneider et al. introduced "FeelCraft" for end user customization of a library of *feel effects* [188].

Kim and colleagues offered combined spatial and temporal control using a tactile video metaphor for dense, regular arrays of tactile pixels ("taxels"), including a feature of sketching a path on video frames [123] (Figure 4.2b). While a promising approach, this tool relies on editing of discrete actuators and frames, with its sketching feature used for input, not as a manipulation method. As well, it does not generalize to sparse or irregular displays, and was not evaluated with designers. We suggest that an animation metaphor could provide an easier interaction model, facilitating key creative activities such as rapid exploration and iteration, especially through a continuous timeline (Figure 4.2c). The control of multi-actuator outputs has also been explored by TactiPEd [169] and Cuartielles' proposed editor [50]. However, these approaches still require the separate control of different actuators, rather than a single perceived sensation produced by the multi-actuator device.

4.4 Tactile Animation Authoring Tool

Our objective is to provide media designers with a familiar and efficient framework for creating dynamic haptic content. Mango's design is based on two sets of requirements: Literature ("LRs", Table 4.1), from prior research on haptic authoring tools, and Industry ("IRs") from interviews with five industry experts in haptic media creation and animation, which confirm and expand upon design decisions for other VT tools.

4.4.1 Gathering Design Requirements

We interviewed two industry experts with haptics experience from a media company (E1-2). E1 uses Max/MSP, OpenFrameworks, Processing, and Visual Studio to create haptic media. E2 is a professional media designer and an expert user of Pro Tools (an industry standard for authoring sound media). Together, E1 and E2 previously undertook a six-month training that included generation of dynamic haptic experiences on seats and supporting platforms using audio and video tools. Our interviews included meetings, recordings, and sketches of their experience during training.

In addition, we conducted contextual interviews of three industry animators (A1-3) interacting with non-tactile animation tools using a think-aloud protocol. A1 and A3 used Adobe After Effects, while A2 used Maya. A1 and A2 were tasked with creating an animation of two balls moving; A3 created an animation

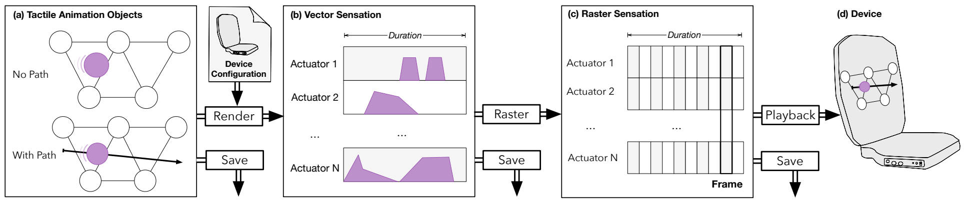


Figure 4.3: Tactile animation rendering pipeline. Users can: (a) create tactile animation objects; (b) render objects to actuator parameter profiles (such as amplitude) with our rendering algorithm; (c) rasterize vector sensations into frames; (d) play the sensation on the device.

based on a sound file. These interviews yielded rich detail that we compiled into categories, then compared with our LRs (Table 4.1). LRs 2-7 also emerged independently from this stage. We extend the LRs with additional expert-drawn **industry requirements (IRs):**

IR1 - Animation window allows users to draw tactile animation objects, control them in space, and define their motion paths. The window is overlaid with location and type of haptic actuators, providing visual feedback (LR8).

IR2 - Timeline is a time track for a tactile animation object. During playback, the animation is played on IR1 showing the movement of the animation relative to the tactile object. Object behaviours are linked to time track to visualize temporal variations. Time tracks are editable by inserting key frames.

IR3 - Object tools extend LR2, supporting direct manipulation operations on tactile objects such as “new”, “scale”, “translate”, analogous to object creation and manipulation in After Effects and Maya.

IR4 - Path tools define motion paths of tactile objects (straight lines, curves, input-device traces), and store them in a path library (LR3).

IR5 - Haptic rendering schemes compute output waveforms for each actuator channel, animated visually in the animation window. Users select the scheme from a list for connected hardware, defined in a hardware configuration file (LR4).

IR6 - Global parameter tools allow the user to control the overall feel of the tactile animation object. Analogous to filters and effects applied on the object, this includes parameter setting for frequency, intensity and modulation.

We developed a tool design from these two sets of requirements. Our Mango prototype uses Python 2.7 and Tkinter for the rendering pipeline (Figure 4.5) and UI (Figure 4.4), which communicates with haptic devices via USB.

4.4.2 Framework for Tactile Animation

In this section, we present an animation metaphor that allows users to generate tactile content in the same way as they would create visual animations and play them real-time on a VT array. Figure 4.5 shows the workflow of this authoring mechanism. Designers create tactile animations on a typical animation tool as shown in Figure 4.5a. The animation object is placed in space, and the designer adjusts its size on the visual outline of the VT array. The designer then adds movements and special effects to the object using Mango's toolset, and plays it to observe its frame-by-frame sequence.

Mango's rendering engine translates visual animations to tactile animations on the VT array. Knowing the location of vibrating points on the sparse array of VT actuators, the rendering engine resolves the animated sequence into individual actuators using the phenomena of phantom tactile sensations [3, 105]. The phantom sensation is a sensory illusion elicited by stimulating two or more vibratory elements on the skin. Instead of feeling the individual vibration points, the user feels a single sensation in between, whose perceived intensity is defined by the weighted sum of the intensities of the vibrating elements. Therefore, in each frame, the animated tactile object is resolved into intensity of actuators on the VT array (Figure 4.5b). The rendering engine then calculates raw waveforms for each VT channel (Figure 4.5c) that can either be sent to the VT device to play the animated sequence or exported as a multichannel datafile for later use. Previous work has interpolated between only two actuators [133, 203]; however, a more generalized 3-actuator interpolation algorithm allows for arbitrary real-time manipulation of the tactile animation object on grid displays.

To accommodate the animation framework, we define three **datatype models**, for use in the current implementation and future expansion of the Mango tool: *Tactile animation objects*, high-level hardware-independent data types for tactile animation; *vector formats*, high-level hardware-specific control common in previous work; and *raster formats*, low-level hardware-specific formats for rendering and playback.

Tactile animation objects are high-level specifications of virtual sensations moving on a 2D VT array (Figure 4.5a). High-level parameters, such as location, size, and other semantic qualities, can either be constant or variable. Each tactile object has a start time and a duration. Object type is also defined for tactile animations that sets pre-defined parameters and features to animated objects. For example, a moving virtual point can have a position, size, and frequency parameter, while a "rain" effect can have a position and more semantic parameters like raindrop frequency or size.

Tactile animation objects are device-independent. Mango uses a device configuration file (LR4) and the rendering engine to create animated VT patterns on hardware. Animation objects can be combined in novel ways, organized in groups, or generate other tactile animations like a particle generator as in a graphical animation tool, and can have paths that constrain motion to a pre-determined trajectory. We prototyped an early version of the tactile animation object in Mango; however, the data type is extensible.

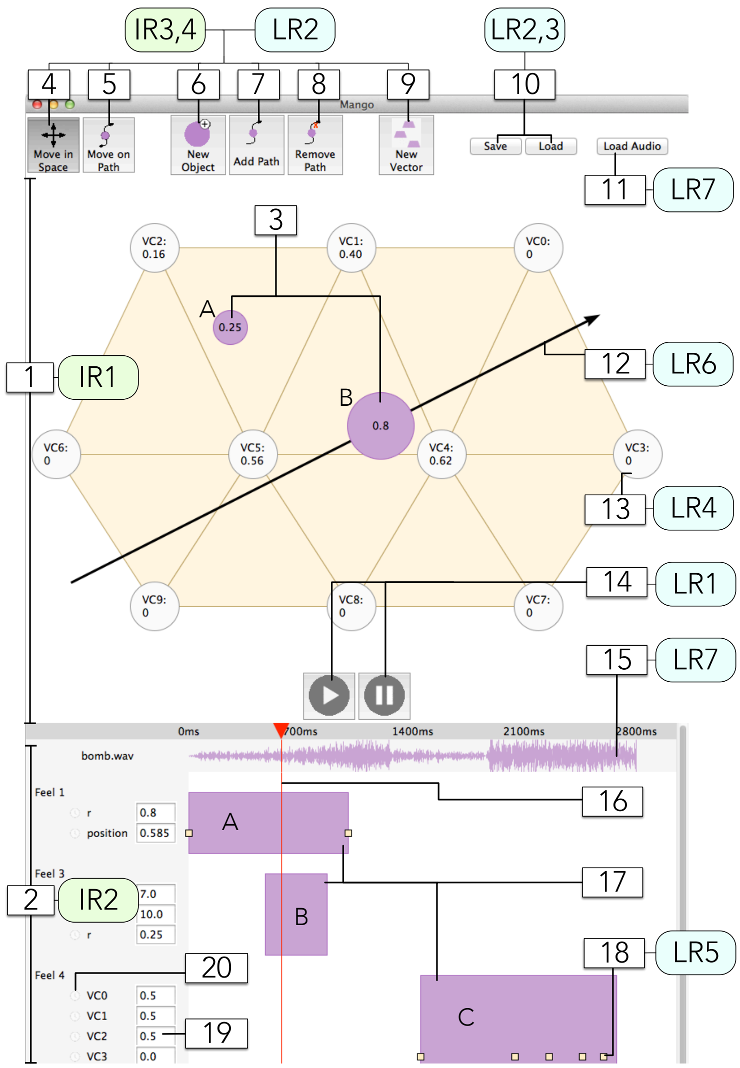


Figure 4.4: Mango graphical user interface. Key components are labeled and linked to corresponding design requirements.

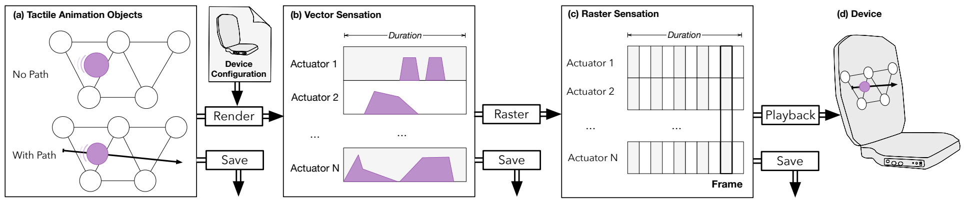


Figure 4.5: Tactile animation rendering pipeline. Users can: (a) create tactile animation objects; (b) render objects to actuator parameter profiles (such as amplitude) with our rendering algorithm; (c) rasterize vector sensations into frames; (d) play the sensation on the device.

Vector formats are similar to those in previous work (e.g., [67]). Instead of objected-based definitions, as in tactile animation objects, parameters are defined for individual actuation. (Figure 4.5b). Parameters include duration, amplitude envelopes (e.g., fade-ins and fade-outs), frequency, and start times. Being device-specific, vector formats offer finer sensation control than tactile animation objects (analogous to pixel-level editing of sprites). However, creating a single percept from independent controls can be challenging. This data type is useful when rendering methods for the hardware are not defined or the user wants to control specific actuator sequence to animate tactile content, such as using the Tactile Brush [105].

Raster format, analogous to a raster-graphics image or WAV file, is suitable for playback operations or exporting it to a device specific format (Figure 4.5c). A raster format contains a matrix of actuator intensities; each row defines intensities of an actuator and columns containing the intensities at each time instance. Each format also contains a timestamp row defined by the rendering engine's framerate. The playback system parses the raster data, finds the current column, and pushes these actuator settings to the device. This data type is also used for real-time feedback during authoring.

4.4.3 Authoring Interface

The authoring interface allows designers to efficiently create moving tactile content in a familiar environment. Here we describe user interactions, most of which are through the animation window (1) and timeline (2) (Figure 4.4).

Animation Window: A user creates a tactile animation object (3) with a "new object" button (6), then manipulates it in the animation window (1). The window is overlaid with a faint trace of the VT hardware (13) for context. Here, we used an array of 10 VT actuators (Figure 4.7).

Object Paths: The animation object (3A) has (x, y) parameters describing position, an “r” (radius) parameter, corresponding to the VT output voltage from 0 (minimum) to 1 (maximum). An optional path can be added to an object (7), or removed (8), along which the motion of the object (3B) is constrained (12). The path-object (3B) is manipulated in two ways: moving on path (5), which moves the object from the beginning (position=0) to the end of the path (position=1), or moving in space (4), which moves the object and the path together on the animation window (1). The current Mango implementation only supports straight-line paths, however their use can be extended in a later version. Also note that curves can be accomplished through keyframed (x, y) positions.

Timeline: Each animation object (3) is represented in the timeline (2) as a track (17). The red scrubhead (16) (shown as a triangle and line) shows and manipulates the current time. Animation objects can be moved in time by clicking and dragging, and resized to change duration. Individual parameters can be set on the left, by typing values into text fields (19), allowing precision. The entire animation can be played and paused using buttons (14) or the spacebar.

Keyframes: Parameters can be toggled as “keyframeable” with a small clock button (20). When the value is changed, a keyframe (18) is automatically created at the current time. Intermediate values are linearly interpolated.

Vector Sensations: A new vector can be created by selecting an object (3) then clicking on a button (9). These sensations control each actuator directly through the parameter values, controlling that actuator’s voltage from 0 to 1 (same as the “r” parameter). The corresponding actuator is highlighted in the animation window (1) when the text field (19) or track (17C) is selected. Each track is also keyframeable.

Save and Load: Animations can be saved and loaded (10) to/from JSON files. An audio track can be loaded (11) to the timeline (15).This allows the user to design a VT experience for sound files (LR7). Video overlay is left for future work.

Hardware Configuration File: A hardware-specific structure is defined and stored in a JSON configuration file (LR4). The file contains: (a) physical width and height of the grid, (b) a dictionary of actuator types (e.g., voice coils or rumble motors), each with a list of control parameters (e.g., frequency, intensity) and allowable values; (c) location and type of each actuator; (d) supported communication protocols and rendering methods; (e) brand information (e.g., USB vendor id and product id) for device recognition; and (f) default settings. Physical dimensions are defined in SI units, e.g., meters, Hz.

Playback: Once the animation of the object is defined, the user can play and stop the animation. During playback, the animation runs in (1) and the corresponding parameters vary in (2). Simultaneously, VT stimulations are activated on the hardware for user feedback. Multiple animation objects and vector sensations can exist simultaneously. Actuators output the sum of all the values generated by objects (described later in the Rendering Algorithm section) and vector sensations.

4.5 Rendering Algorithm

Mango's rendering algorithm defines how high-resolution haptic feedback is translated to sparse grids of VT actuators. The rendering algorithm translates animations created in the animation window to animated VT patterns on the hardware. Figure 4.5 shows the rendering pipeline that converts animation objects to a raster format, which outputs to the hardware.

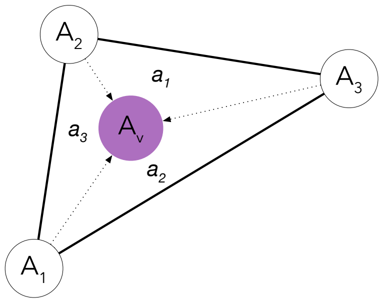
The rendering algorithm is derived from psychophysical understanding of VT illusions on the skin and creates percepts of virtual actuators and their motion in between a set of real actuators. The precise perceptual model depends on several factors, such as type of VT actuators (DC vs. voice coil motors), stimulation site (forearm vs. back) and the spacing of actuators in the array (e.g., [105]). To allow for custom framerates and real-time feedback, we generalize from the 1D case (in between two VT actuator along a line) to the 2D case (in between three or more actuators, previously accomplished with non-VT sensations [220]). Thorough investigation of the psychophysical model is beyond our present scope, however, we empirically determine the most effective model among those documented in the literature for the 1D case with a pairwise comparison.

4.5.1 Perceptual Selection of Interpolation Models

The rendering algorithm translates virtual percepts to a physical actuator grid. We first construct a Delaunay triangulation for all actuators to automatically define a mesh on the hardware grid. At each instant of rendering, we use barycentric coordinates of the virtual animation objects relative to a triangle defined by three real actuators (Figure 4.6a). Barycentric coordinates are scaled by an interpolation method to determine real actuator intensity.

We propose three interpolation models for Mango, derived from prior psychophysical understanding of phantom VT sensations: (i) *linear*, (ii) *logarithmic ("log")*, and (iii) *Pacinian power ("power")* (Figure 4.6b).

In the linear interpolation model, barycentric coordinates are linearly related to actuation amplitude. In the log model, these coordinates are scaled logarithmically, as perceived intensity is related to physical vibration amplitude [226]. In the power model, coordinates are coupled to the power (square of the amplitude) of vibrating stimulations [226]. Linear and log interpolation models have been used in the past to express either location or intensity respectively (but not both) of virtual sensations between two vibrators [3, 203]. A Pacinian power model was used in [105] to account for both location and intensity of virtual sensation between two vibrators.



(a) Barycentric coordinates

Linear $A_i = a_i \times A_v$

Log $A_i = \frac{\log a_i + 1}{\log A_{max} + 1} A_v$

Power $A_i = \sqrt{a_i} \times A_v$

(b) Candidate interpolation methods

Figure 4.6: Interpolation models to determine physical actuator output (A_{1-3}) from virtual actuator intensity (A_v) and barycentric coordinates (a_{1-3}).

4.5.2 Pairwise Comparison Study

To determine the preferred model for this VT hardware in Mango's rendering pipeline, and to identify relevant factors (e.g., frequency, amplitude), we performed a pairwise comparison of our three candidate interpolation models.

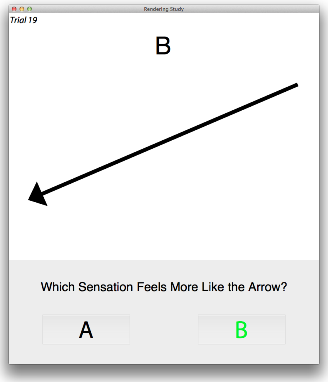
Participants and Apparatus

Eighteen volunteers took part (6 female, between age 20-35). The VT hardware consisted of 10 high-quality VT actuators (C2 tactors, Engineering Acoustics, Inc., USA) arranged in a 3-4-3 layout and mounted on the back of a chair in a pad 21 cm high, 29 cm wide, and 2 cm thick; actuators form equilateral triangles with edges of 6.35 cm (Figure 4.7b). The rendering engine updates at 100 Hz. Through piloting, we determined that the device's on-screen visual outline should mirror the sensations rendered on the physical device. That is, if participants see an animation object on the right side of the screen, they prefer to feel it on the right side of the back. Figure 4.7a shows the experiment interface, in which an arrow represents the sensation direction.

Methods

We conducted A/B paired comparison tests (two-alternative, forced-choice) to determine the preferred model out of the three candidates. In each trial, participants were presented with two stimuli at a 400 ms interval. Each stimulus is a "straight-line" VT stimulation on the back using one model. Participants were asked to select the stimuli that *best represented straight-line motion* in a variety of directions.

Two durations (500 and 1500 ms), eight cardinal directions, and A/B order



(a) Rendering study interface

(b) Output device with highlighted actuators

Figure 4.7: Rendering study setup and user interface.

were crossed with each model pair, and presented in a random order. For each trial, frequency was randomly selected from 80, 160, 240, and 300 Hz, and intensity from between 10 and 20 dB above detection threshold. Each participant performed 96 trials over ∼15min (1728 total).

Results

Each algorithm pair's data was fit to a logistic regression model with participant, frequency, intensity, direction, and duration as factors; direction was grouped into horizontal, vertical, and diagonal. We performed stepwise regression (backwards elimination with $\alpha = 0.05$ and a χ^2 test for removing each factor) to iteratively eliminate factors that were not statistically significant.

Logarithmic vs. Linear. Regression eliminated duration, frequency, intensity, and direction ($p > 0.1$). The resulting model has Nagelkerke $R^2 = 0.135$. Using Bonferroni correction for multiple comparisons, 95% confidence intervals for each participant were computed. 11 participants were more likely to prefer Log over Linear ($p < 0.05$) models; none were likely to prefer the Linear model.

Logarithmic vs. Pacinian power. All 5 factors were eliminated ($p > 0.1$). The overall 95% confidence interval of participants selecting Log over Power was 37.06% to 87.40%, overlapping 50%. We therefore detected no significant difference of preference between Log and Power models.

Pacinian Power vs. Linear. We eliminated intensity, direction and duration ($p > 0.1$), with the fitted model's Nagelkerke $R^2 = 0.0970$. The confidence interval for each participant-frequency combination, via Bonferroni corrections, yielded 22

/ 72 participant-frequency combinations selecting Power model over Linear model more than 50% of the time. No one chose the Linear model more than 50% of the time.

Conclusion: Logarithmic interpolation outperformed linear and was equivalent to Pacinian power model. We proceeded with the logarithmic model for Mango's implementation, as the power model did not outperform either of the others.

4.6 Design Evaluation

To evaluate Mango's animation metaphor and expressive capability, we asked media professionals to create a variety of designs. Qualitative evaluation was chosen for rich, focused, early feedback of the animation metaphor and lessons for iteration. A quantitative comparison between tool perspectives is left until more refined tools are developed. We wanted to establish whether this is an effective approach before studying the most effective approach.

Six participants (P1-6, 3 females) were introduced to Mango driving the VT hardware described previously. P1 had experience with haptics but not animation beyond video editing; P2-5 had animation experience but little or no experience with haptics; P6 had no experience with haptics or animation, but was familiar with media tools like Adobe Photoshop. P5 was also involved with the requirement gathering interviews presented earlier. Each entire session took 40 to 60 minutes.

Each participant was introduced to Mango with a training task: designing an alerting sensation using either animation objects or vector sensations (order counterbalanced). Then, each participant was given three design tasks. 1) Primarily *temporal*: create a heartbeat sensation. 2) Primarily *spatial*: tell a driver to turn left. 3) *Context-based*: create a tactile animation to match a sound file. A 3-second sound effect of a bomb falling (with a whistle descending in pitch) then exploding with a boom was chosen, i.e., complex with two semantic components. The wide array of resulting designs can be found in the accompanying video. Mean non-training task time was 5:59 (med 5:38, sd 2:46, range 1:41-13:48).

After each task, participants rated confidence in their design from 1 (Not confident) to 5 (Very confident), primarily to stimulate discussion. All designs were rated 3 or higher; P6 wrote "6" for his sound-based design. The animation object training task was always rated the same or higher than the corresponding vector training task. While suggestive, these ratings were self-reported and from a small sample. We thus did not conduct statistical analysis.

A semi-structured interview followed the design tasks. Participants were asked to compare animation objects with vector sensations, and to walk through the interface to elicit feedback. Interviews were conducted and analyzed by a re-

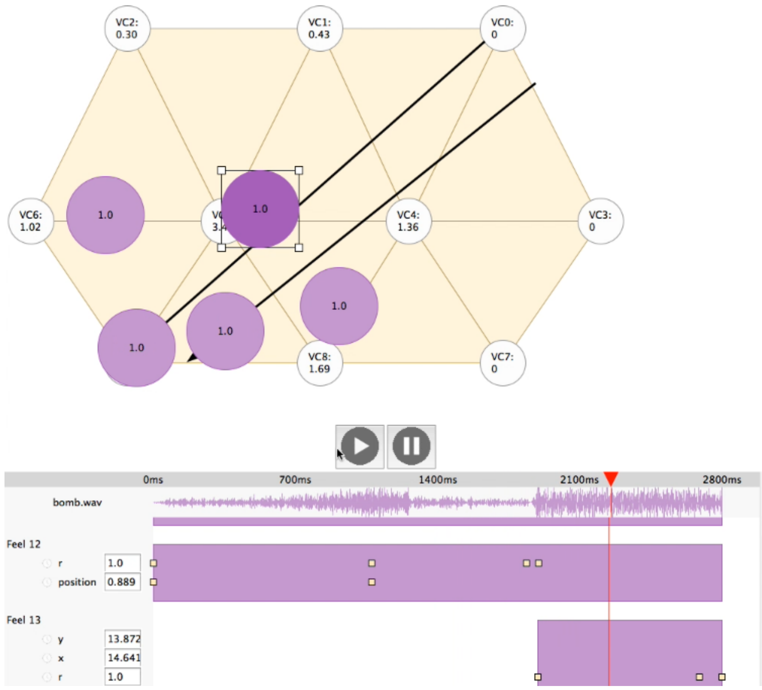


Figure 4.8: Example of P2's animation for matching a sound. See the accompanying video for all participant animations.

searcher with training and experience in qualitative research, and followed established methodologies: methods of grounded theory [44] informed by phenomenological protocols [157]. Analysis resulted in four themes.

Theme 1: Animation Metaphor

Participants found the tool easy to use. All six participants were able to accomplish all five tasks (object alert, vector alert, heartbeat, turn left, sound). Participants described the interface as intuitive (P1-5), agreeing that it was an animation tool: *"It's up to the standards of other animation tools"* (P1), *"This is totally animation"* (P2), *"It felt very much like an animation tool"* (P4), *"I'm not an expert when it comes to haptics, but this software seems almost as if it can change the game of designing haptic vibrations"* (P5). Negative feedback focused on polish and feature completeness: *"gotta spline [the keyframe interpolation]"* (P2), *"a couple quirks but there was nothing difficult to overcome"* (P4), *"being able to design your own curve [path] would be really nice"* (P5).

Theme 2: Tactile Animation Object vs. Vector Sensations

Participants relied more on animation objects than vector sensations, which were only used twice: P4's heartbeat task and P5's sound task (combined with an animation object). P1 switched from vectors to animation objects early in her heartbeat task; no other participants used vector sensations.

Animation objects were described as easier to use and more intuitive, especially to represent location or for non-animators. *"After using the new object I'd probably never use new vector again"* (P2), *"easier to find the location of the heart"* (P1), *"if I weren't an animator I think I would only use [animation objects]"* (P4). Vectors were preferred for more fine-tuned control when motion didn't matter as much, often using many keyframes. *"You can control multiple [actuators] at the same time, so you don't have to create new objects and then put them everywhere on the screen"* (P1), *"[Animation objects] can be more comfortable to use when one doesn't work with keyframes"* (P3), *"If you want precise control over [actuators], then vector is the way to go"* (P4).

Theme 3: Designing-in-action with direct manipulation

Participants used direct manipulation to feel their designs in real time, dragging animation objects and scrubbing through the timeline: *"I would make the [animation] object and just play around with it before creating the animation, as a way to pre-visualize what I was going to do"* (P5), *"I kind of play around with it, and randomly come up with the ideas"* (P6). P2 even noted that YouTube did not have real-time video scrubbing feedback like Mango's: *"I wish I could scrub back and forth [with YouTube]"* (P2). However, continual vibrations were annoying, and participants requested a "mute" feature: *"It would be nice if...it doesn't go off constantly."* (P3).

More generally, participants used feedback from their experience or external examples. P1 stopped to think about her own heartbeat, P2 used a YouTube video of a heartbeat as a reference, and P3 based her alert on her phone: *"It's typical to have two beeps for mobile phones"* (P3). Correspondingly, participants were excited when prompted by an audio sensation: *"I was really happy with the bomb one, because I could really hear it and imagine me watching a TV and then feel it at the same time"* (P1), *"The sound part was good, that would be a fun thing to design for"* (P4).

Theme 4: Replication through Copy and Paste

Replication in both space and time was common while using Mango. Many designs had symmetrical paths to reinforce sensations (Figure 4.8). All but P4 requested copy / paste as a feature. *"I could just copy/paste the exact same thing on the left side and then move it to the right side"* (P1), *"I have the timing the way I like it, ideally it'd be cool if I was able to copy and paste these, so it would be able to repeat"* (P5).

4.7 Discussion

Here we interpret our design evaluation, explore animation with other devices, and describe applications and limitations.

4.7.1 Design Evaluation Summary

From our design evaluation, we conclude that tactile animation is a promising approach for controlling tactile grids. Direct, continuous manipulation of tactile animation objects supported embodied design and exploration by animators, who rapidly iterated on designs to try new ideas. Mango facilitated the design of a wide variety of animations (see accompanying video) and received positive responses. We also found recommendations for our next iteration: more animation features, video as well as audio context, and muting.

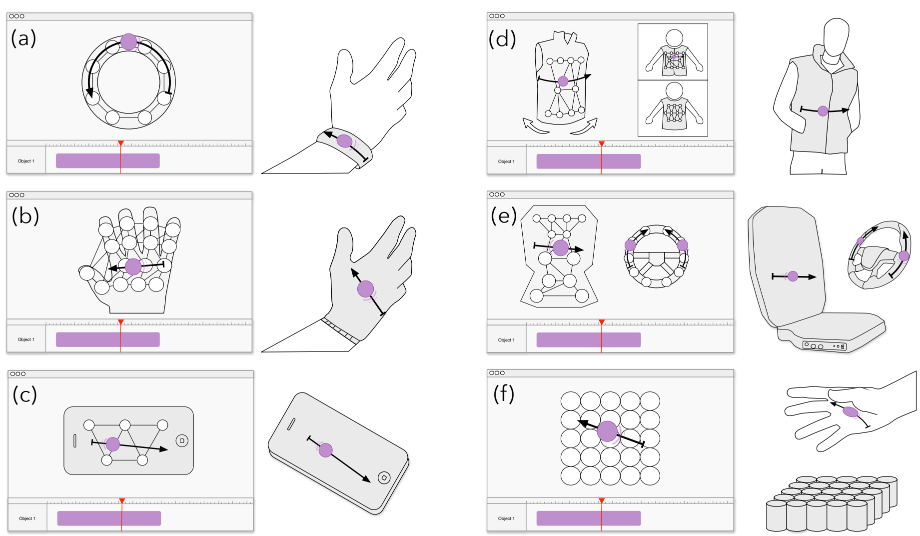


Figure 4.9: Tactile animation could define motion with (a) 1D actuator arrays, (b) dense and sparse VT grids, (c) handhelds, (d) 3D surfaces, (e) multi-device contexts, and (f) non-VT devices like mid-air ultrasound.

4.7.2 Possible Extension to Other Device Classes

The animation metaphor is not limited to a back-based pads. Part of the advantage of an abstracted animation object is that, as long as a suitable rendering algorithm can be developed, the metaphor can apply to other devices. In this section, we illustrate possibilities that we plan to explore in future work.

1D VT Arrays (Figure 4.9a): 1D VT arrays are common in arm sleeves, wrist bands, belts, and similar wearables. These devices provide sensations along the path of the array. By constraining objects to a linear or circular path, barycentric coordinates collapse into 1D interpolation.

Dense and Sparse VT Grids (Figure 4.9b): 2D VT grids are also common, used in chairs, gloves, and the backs of vests. While we evaluated Mango with a sparse back-mounted array, tactile animation naturally supports denser arrays, either with our rendering algorithm or by using a nearest-neighbour technique to activate a single actuator.

Handhelds (Figure 4.9c): Actuators embedded in handheld objects, such as mobile devices, game controllers, or steering wheels, shake objects instead of directly stimulating the skin. Animators might be able to define source locations for vibrations using handheld-based rendering algorithms (e.g., [203]).

3D Surfaces (Figure 4.9d): Mango currently only supports a 2D location for its animation objects. However, tactile animation can be extended to support surfaces of 3D surfaces, such as vests or jackets that wrap around the user's body. More work will need to be done to perfect this interaction style, possibly using multiple views or a rotatable 3D model with animation objects constrained to the surface.

Multi-device contexts (Figure 4.9e): Mango's rendering algorithm already supports connections to multiple devices simultaneously. The editing interface could combine layouts for different devices, enabling animators to animate the entire user experience (such as a car's seat and steering wheel).

Non-vibrotactile devices (Figure 4.9f): While our rendering algorithm is particular to VT arrays, a tactile animation object can represent manipulable percepts with other actuation technologies. Ultrasound-based mid-air displays generate a sensation as a focal point with a position and size [231]; this sensation could be manipulated through a tool like Mango. Similarly, passive force-feedback sensations (e.g., Hapseat [53]) or height displays (a grid of pins) could be supported.

4.7.3 Interactive Applications

While our goal was to enable animators to create rich content, the tactile animation object can be linked to alternative input sources for other interactive experiences.

User gestures. User gestures and motion can be tracked and mapped to animation objects directly rendered on the haptic hardware. For example, a user creates patterns on a touch sensitive tablet that maps touch locations to a grid. Users could play games or create personalized haptic messages on the back of a vest. Similarly, a dancer's movements could be tracked through accelerometers, drawing animated haptic content on the body of her audience through actuated theater seats during a live performance.

Camera feed extraction. Motion from video feeds can be automatically extracted with computer vision and rendered on grid displays [122], providing dynamic patterns associated with actions during sports, movies, and games. Similarly, animation parameters could be extracted and mapped to positions on a VT grid, creating haptic feedback for non-haptic media.

Data streams. One main application of haptic grid displays is to provide users directional, assistive, and navigational cues during driving cars, walking down the street, or with over- saturated sensory tasks. Users could associate digital data streams, such as GPS input, to predefined set of directional patterns on the back or palm of the hand.

4.7.4 Limitations

While the tactile animation metaphor seems promising and may apply to many contexts, it is limited by the requirement of a suitable rendering algorithm for target hardware. We have not yet explored other form factors, such as handhelds, multi-device scenarios, or non-vibrotactile sensations. Although we perceptually optimized our algorithm, we did not conduct a full psychophysical investigation. Further work needs to be done to identify the limits, thresholds, and peculiarities of this rendering technique. Examples include: curved trajectories of animation objects (although participants' use of curved motion was encouraging, e.g., P5's turn left sensation), spatial frequency control (how to superpose animation objects of differing frequencies), non-triangular meshes (e.g., quadrilateral interpolation or kernel methods), and mixed actuator types (such as a chair with both voice coil and rumble motors, Figure 4.9e).

4.8 Conclusion

This paper introduces the *tactile animation object*, a new abstraction for creating rich and expressive haptic media on grid displays. This animation metaphor allows designers and media artists to directly manipulate phantom vibrotactile sensations continuously in both space and time. Our rendering pipeline, which uses a perceptually-guided phantom sensation algorithm, enables critical real-time feedback for designing. We incorporated these ideas into a prototype, Mango, with a design grounded in animator requirements and haptic design guidelines. Professional animators used our tool to create a variety of designs, giving positive feedback and excitement for future versions. This approach has the potential to accommodate a large variety of haptic hardware, ranging from a single shaking element mounted on the seat to an array of actuators stimulating multiple points on the skin, and can export content into formats applicable in the production pipeline. Tactile animation empowers animators with a new set of artistic tools for rich, multimodal feedback.

4.9 Acknowledgments

We thank our reviewers and participants for their valuable feedback. This work was supported by Disney Research, with additional support provided by NSERC.

Chapter 5

Browse: Macaron

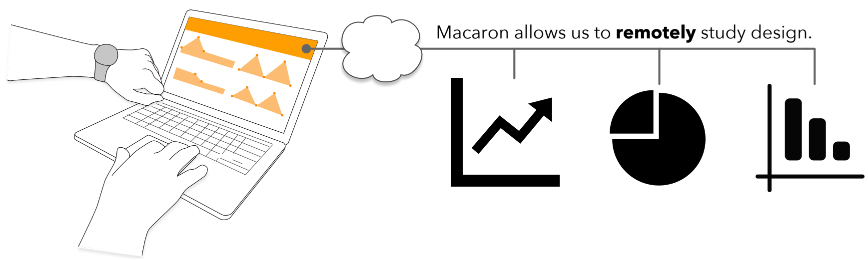


Figure 5.1: Concept sketch for a Macaron, an online, open-source VT editor features incorporable examples and remote analytics.

Preface – In our third vibrotactile design tool, Macaron[1], we explored the design activity of *browsing* external examples. Because we explored HaXD tool implementation in-depth in Chapter 4, we knew how to build Macaron; we thus focused on studying the design process, specifically, how different ways of viewing or reproducing elements of a vibrotactile icon affects design. We refined our study tasks: participants designed haptic tracks for visual animations. We based this task on the effective sound-based task in Chapter 4. We used phenomenology and grounded theory methods augmented by logged user actions and visualized timelines to look at our participants' design process; our participants were generally naïve to haptics and media design to complement our previous studies. We saw the different stages of design, including *browsing*, *sketching*, and iterative *refinement*.

[1]Schneider and MacLean. (2016) *Studying Design Process and Example Use with Macaron, a Web-based Vibrotactile Effect Editor*. HAPTICS '16: Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems.

5.1 Abstract

Examples are a critical part of any design process, but supporting their use for a haptic medium is nontrivial. Current libraries for vibrotactile (VT) effects provide neither insight into examples' construction nor capability for deconstruction and re-composition. To investigate the special requirements of example use for VT design, we studied designers as they used a web-based effect editor, *Macaron*, which we created as both an evaluation platform and a practical tool. We qualitatively characterized participants' design processes and observed two basic example uses: as a starting point or template for a design task, and as a learning method. We discuss how features supporting internal visibility and composition influenced these example uses, and articulate several implications for VT editing tools and libraries of VT examples. We conclude with future work, including plans to deploy Macaron online to examine examples and other aspects of VT design *in situ*.

5.2 Introduction

Creativity often sparks when an inventor, examining existing ideas, sees a way to combine them with a novel twist [227]. An environment rich with *examples* is fuel for this fire. In industrial and graphic design [27, 97] their use improves process and final results [60, 129].

Several effect libraries are available to designers of vibrotactile (VT) sensations, e.g., for accessible wayfinding [**?**] or media experiences [51, 108, 188**?**]. But despite the need for effect customizability [200], VT library elements are generally opaque in construction and immutable. Recent advances include limited parameter adjustability [108, 188] and faceted library search and browsing [201]. Despite this, designers still must either choose a pre-existing sensation or build from scratch: *elements cannot be sampled, recombined, built upon or adapted*. In contrast, web designers can access a page's source; graphic and sound designers can sample and incorporate colours and sounds from other media.

Here, we *examine the potential role of examples* in VT design, to establish how to best support their use. We designed a web-based editor and interactive *design gallery* [129, 146] (Figure 5.2) for VT sensations, then asked users to compare versions (Figure 5.3) that vary in example accessibility via *visibility* and *incorporability*, as they create VT effects for animations (Figure 5.4).

Analysis of user action logs provide an objective picture of the VT design process. To validate the deployment of this methodology at scale, we also interpret and validate logs with direct observation and interviews. Specifically, we:

- introduce *Macaron*, a web-based VT effect editor through which examples can be used directly in designs,

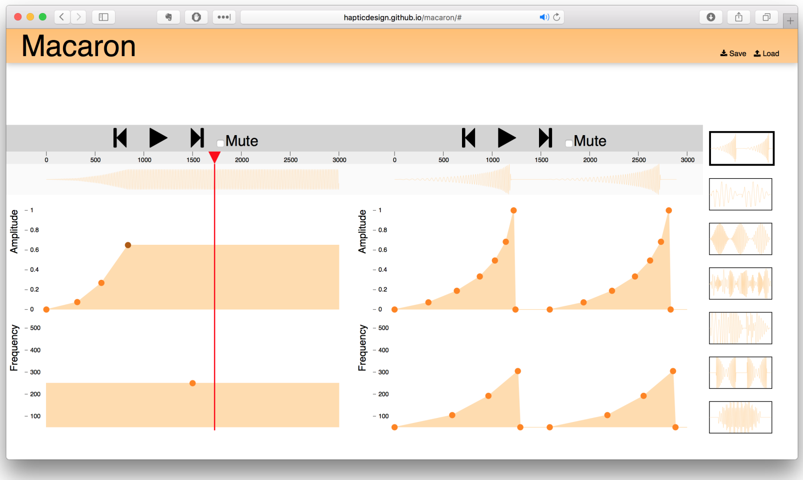


Figure 5.2: Macaron interface, “`hi`” version featuring both composability (copy and paste), and visibility of underlying parameters. The user edits her sensation on the left, while examples are selected and shown on the right. Macaron is publicly available at `hapticdesign.github.io/macaron`.

- find that *visible, incorporable examples make design easier* by providing a starting point for design and scaffolding to learn how to work with VT parameters,
- identify *implications for future tools and libraries*, and
- discuss the *opportunities afforded by a web-based editor* as a practical tool and platform for studying other aspects of VT design at scale.

5.3 Related Work

5.3.1 Salient factors in VT effect perception and control

Vibrotactile effects (e.g. haptic icons [140]) are typically manipulated with low-level engineering of signal parameters, beginning with amplitude, frequency and waveform [17, 88, 140**?**]. Rhythm can support large, learnable icon sets [221**?**]; combining waveforms enhances roughness [**?**]. Time-varying amplitude adds musical expressivity, from tactile crescendos [22] to envelopes [191]. Multi-dimensional scaling can be used to identify and elaborate these parameters [68, 140**? ?**].

Affect and metaphor are another way to structure and manipulate sensations at a level more cognitively relevant than engineering parameters. Perceived valence (pleasantness) and arousal can be influenced by frequency/amplitude combination [163, 239]. Metaphors [162, 201**?**] and use cases [201**?**] offer structure, memorability and design language. Spatial displays require additional controls for location and direction, whether body-scale [88, 105], mobile [203], or mid-air [163]. While many parameters are available for VT design, we chose the most established (time-varying frequency and amplitude) for Macaron's initial implementation.

5.3.2 Past approaches to VT design

Past editors – e.g., the Hapticon Editor [67], Haptic Icon Prototyper [215], posVibEditor [184], Vivitouch Studio [216], and Haptic Studio (www.immersion.com) – are track-based, with graphical representations to edit either waveforms or profiles of dynamic parameters. Additional features (e.g., spatial control or mobile interfaces) are surveyed in [196].

A library of effects is critical for haptic design tools [196]. Most existing tools support feature saving/loading, and some have an internal component library [67, 215, 216]. However, previous implementations were primarily *compositional*, employing building blocks [68] rather than complete artifacts. Example use was not studied.

Large VT libraries contain complete artifacts, but impose a serious constraint on their use. In the Immersion Touch Effects Studio library, underlying structure and design parameters are hidden and cannot be incorporated into new designs. VibViz [201] features 120 VT examples with visualizations searchable by several taxonomies, but the selection model is all-or-nothing. FeelCraft [188] proposes a community-driven library of feel effects [108] for simple parametric customization and re-use. While end user customization-by-selection is important [200], experts need a more open, editable model, just as web designers rely on full access to source code with recent tools allowing search and easy incorporation [129].

5.3.3 Examples in non-haptic design

Problem preparation – also known as the "problem setting" [198] or "analysis of problem" [227] step of design – involves immersion in the challenge and drawing inspiration from previous work. Both may come from the designer's experience, *repertoire* [198] or exposure to a symbolic domain, e.g., mathematical theorems and notation [49].

To this end, external examples are critical in inspiring, guiding and informing design [27, 97]. Industrial designers collect objects and materials; web designers bookmark sites [97]. In graphics and web design, *design galleries* organize examples to be immediately at hand [129, 146]. Example-based tools often use sophisticated techniques to mix and match styles and content [**?**]: this requires immediate access to the examples' underlying structure.

5.4 Apparatus Design

To investigate VT design in the context of examples, we required a platform that would expose users' natural procedural tendencies. Our Macaron design gallery is simple, flexible, and extensible. In this work, we add multiple types of example access to polished implementations of familiar concepts: *tracks*, *envelopes*, and *keyframes* (Figures 5.2,5.3).

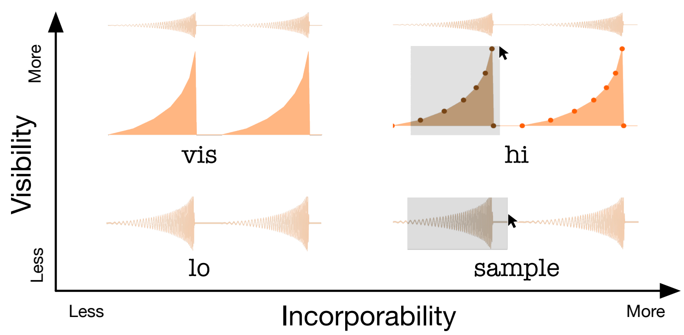


Figure 5.3: Design space for Macaron versions. `hi` and `sample` both allow for selection and copying of example keyframes. `vis` and `hi` both show the underlying profiles. `lo` represents the current status quo; only a waveform is shown.

Tracks are the accepted language of temporal media editors (video, audio, and past haptic efforts [67, 184, 215]). We provide tracks for perceptually important "textural" parameters (amplitude and frequency); the user accesses periodic and time-variant aspects by manipulating their *envelopes* using *keyframes*, with linear interpolation in-between. Users double-click to create a new keyframe, click or drag a box to select, and change or delete a selection by dragging or with the keyboard. A waveform visualization reflects changes.

Macaron's example access features are inspired by more recent graphics and web design galleries [129, 146**?**], which show examples side-by-side with the editor. Other implemented features, critical for polished creative control [196], include real-time playback, time control (scrubbing) copy-and-paste, undo and redo,

`hi`	Full access to gallery examples, with keyframes visible and selectable for copy and paste. Simulates source visibility, *e.g.*, viewing the source of a web page or having access to a `.psd` PhotoShop document.
`sample`	Hides underlying parameters of frequency and amplitude, whereas waveform regions (underlying keyframes) may be copied and pasted into a design, simulating example mixing in absence of visibility into underlying construction. While possible to see underlying representation by copying the entire example, the steps are indirect and inconvenient.
`vis`	Reveals underlying parameters, but hides keyframes, parameter scales, selection and copy/paste features. The inverse of `sample`, it exposes example structure, but does not support incorporating example elements into a design.
`lo`	Supplies a "black box" outer representation. Playback and visualization of the complete vibration reflect the status quo of non-visible, non-mixable example libraries.
`none`	No examples present.

Table 5.1: Macaron tool alternatives, varied on dimensions of internal visibility and element incorporability.

and muting (disables realtime VT output). To support its use as an experimental tool, user interactions are logged; start / stop buttons allow the user to indicate when they began and completed their design process.

Macaron was built with HTML5 and JavaScript, using React, Reflux, D3, and Audiolet[2]. Real-time sound synthesis drove a C2 actuator. To leave hands free for keyboard and mouse, the C2 is attached to a wristband; we simulate the design process for a wrist-worn wearable (as in [201]).

Evaluation Versions: To study how examples impact design, we made four gallery versions by sampling two theoretical dimensions of example access: element *incorporability* and internal parameter *visibility* (Figure 5.3, Table 5.1). We hypothesized these would affect users' design processes, e.g., incorporable examples would encourage "mixing and matching" of examples, visibility might provide insight.

We compared these versions with each other and with a non-example version: `none`. In all versions with examples, the user can play or scrub the example, feeling it and seeing the waveform visualization. We did not allow users to modify the examples, to avoid study workflow confounds. To populate the gallery, we

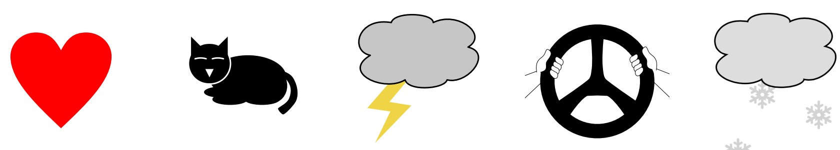
[2]facebook.github.io/react, github.com/reflux, d3js.org, github.com/oampo/Audiolet

chose or adapted seven examples from [201], piloted them to confirm example variety, then regenerated keyframed versions with Macaron.

5.5 Study Methods

Participants were tasked with creating a sensation to accompany five animations (Figure 5.4) – SVGs (scalable vector graphics) which can be played or scrubbed by the same means as navigating Macaron's time control. We chose animation variety (concrete to abstract) and complexity to inspire non-obvious solutions without overwhelming.

Participants were first trained on `none` with no animation, then presented with five animation/version combinations. As the least crucial source of variance, animations were presented in Figure 5.4's constant order, while interface versions were counterbalanced in two 5x5 Latin square designs. Thus, each participant encountered each animation and each interface version once; over all participants, each animation/version combination appeared twice, with Latin squares balancing 1st-order carry-over effects. This design confounds learning with animation task. We believe this is an acceptable tradeoff at this stage, allowing us balance interface order with a single participant session of reasonable length (1-1.5h).



(a) Heartbeat. **(b)** Cat. **(c)** Lightning. **(d)** Car. **(e)** Snow.

Figure 5.4: Animations used as design tasks, in presentation order. Heartbeat expands in two beats; the cat's back expands as breathing and purring; lightning has two arhythmic bolts; the car oscillates up and down, and makes two turns: left then right; snow has three snowflakes float down.

5.6 Results

We targeted a study size of 10 complete participants for a balanced Latin square design, and a manageable sample size for rich, exploratory, qualitative analysis. 13 untrained participants were recruited: P1-10 (7 female, ages 22-35) completed all five tasks, while I1-3 (2 female, ages 29-45) only completed the first three due to time restrictions. Because I1-3 (and P9) all had the same interface order (`lo`, `none`, `vis`, `hi`, `sample`), we suspect that beginning with 'sparse' versions gave

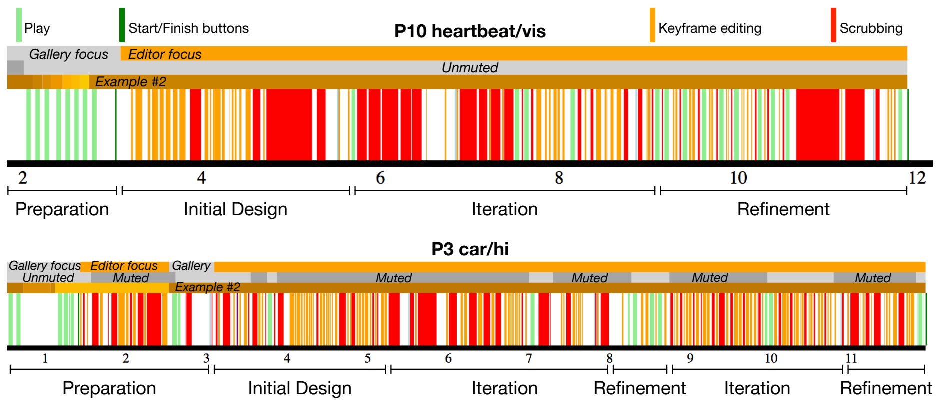


Figure 5.5: Log visualizations showing archetypal design process. Top: P10's heartbeat/`vis` condition (an"ideal" version). Bottom: P3's car/`hi` condition (variations: a return to example browsing after editing, repeated refinement, muted editing).

insufficient insight into how to design quickly enough to finish the study. I1-3 showed no distinct patterns beyond this; we leave their data for future analysis.

Analysis and Data: A team member trained in qualitative methods analyzed screen recordings, interviews, and logs with grounded theory methods (memoing, open & closed coding [44]) and thematic analysis and clustering [157]. We visualized logs using D3 (Figure 5.5). We chose a qualitative analysis because our goal was to capture the design process, not compare Macaron with previous tools. Our analysis exposed three major qualitative findings, discussed below.

Tool Usability: Overall, the tool was well received, described as *"easy to use"* (P1), *"well made"* (P5), *"pretty neat"* (P9), *"the templates help a lot"* (P3).

Completion time: Overall mean task completion time for P1-10 was 5m48s (median 4m48s, sd 3m52s, min 40s, max 18m23s). We conducted two one-way ANOVAs on completion time; neither interface ($p = 0.87$) nor task ($p = 0.64$) had a significant effect.

5.6.1 Archetypal Design Process

Log visualizations (Figure 5.5) show that users could and did employ Macaron for all key design stages: preparation, initial design, iteration, and refinement. All participants followed this sequence. Some omitted one or more steps depending on personal style and strategies for using examples (below). We list observations of the basic process in Table 5.2, to document behaviour and frame discussion.

Prepare	All participants began with a problem preparation step [227]. They played the animation to understand the problem, then typically looked at several (sometimes all) examples. Only P2, P8, and P9 had a task where they did not begin with an example. Otherwise, participants browsed examples, chose a best match to the animation (*"I was trying to find the best match with the visual"* (P7, heartbeat/`hi`)), then transferred into initial design. Participants rarely returned to examples for more exploration; only P3 (car/`hi`) and P5 (car/`lo`) switched to a different example after beginning their initial design. Preparation is characterized by a large number of plays and example switches: on average, 47.45% of all session plays were before the first edit (sd 30.15%), and participants switched examples an average of 6.75 times (sd 5.17).
Initial Design	Participants either used their example choice to help create their initial design, or ignored it because it wasn't close enough to what they wanted to do. Participants typically recreated the example in their editor by copy/paste of the entire design (P1,2,4-8,10) or sometimes a component (P3,10) in incorporable conditions (`hi` and `sample`), or by manually recreating the design (P5,6) or a component (7,10) with `vis`. In the `lo` condition, we only observed P5 somewhat recreating an example. Occasionally, participants would create a new design loosely based on the example rather than recreating it (P3,4,6-8), when using the *Inspire* example use strategy (described later).
Iterate	Participants refined designs with longer periods of editing typically book-ended by playing the entire design (discussed as "real-time feedback" micro interaction pattern). In some cases, especially when the example was "close enough", participants skipped iteration (*Adjust* or *Select* example use strategies, described later).
Refine	Smaller changes forecast design conclusion, e.g., incremental global changes: constant frequency (P1,2,5,6,10), alignment (P1,3,6), or pulse height adjustment (P1,3,8,10). This step is sometimes visible in activity logs, as most participants (P1,3-10) exhibited more frequent plays of the entire design, and shorter periods of editing/scrubbing. Occasionally, participants repeated larger iterations and refinement (P3 car/`hi`, Figure 5.5).

Table 5.2: Steps in observed archetypal design process.

5.6.2 Micro Interaction Patterns Enabled by Tool

Several small-scale patterns further characterize behaviour within the archetypal process.

Different paths through the interface – We saw three design-path strategies.

– *Time* (Figure 5.6a; P1,2,3,4,7,9): proceed through the timeline, creating amplitude and frequency at the same time.

– *Component* (Figure 5.6b, P1,4,6,8,10): iterate on a design element, then repeat or copy/paste it later in time.

– *Track* (Figure 5.6c, P2,3,6,7,8-10): proceed through one entire *track* (typically amplitude), then the other one.

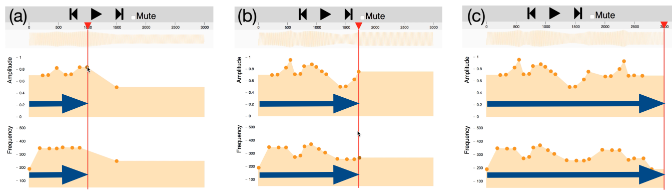
Strategies were often combined hierarchically. P6 developed a car/`lo` component by track (amplitude, then frequency). Wanting additional flexibility, P1,3,7 requested copy/paste *between* tracks: *"The one thing I found missing was copy and pasting between amplitude and frequency"* (P7).

Further showing diverse workflows, participants requested more powerful controls to work with keyframes as a group, such as widen (P5), reverse (P7), shift everything (P9), move up/down and smooth (P4). Other requested features include looping (P1), hovering over a point to see the value (P1), more detail through a zoomable interface (P4).

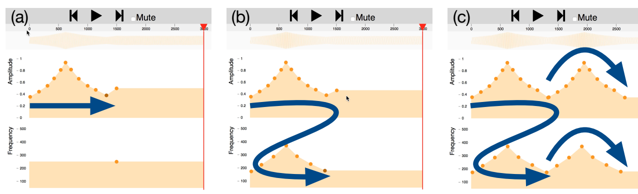
Alignment and Copy/Paste are Precise, Convenient – Precision was valued; alignment and copy/paste were used to achieve it. Alignment was sought both in time and to keyframe values. A common technique (Figure 5.6ab) was to use the red playhead like a plumb-line to align keyframes with animation features (P1-5,7,9,10) and between the two tracks (amplitude and frequency) (P3-5,7,9,10): *"Using that red arrow thing and placing the dots when it makes the heartbeat"* (P2). Some participants, including those who used the plumb-line, requested more refined alignment features: *"I couldn't keep it straight"* (P1).

Copy/paste was used for improved work efficiency (especially helpful during initial layout or when creating long or repeating designs) and precision: *"Copy and paste...was also the most precise, because if you feel like it's a perfect fit, you can use it exactly"* (P6). Correspondingly, conditions without copy/paste (*i.e.*, `lo` and `vis`) took additional effort: *"It's harder...because there's no copy and paste"* (P5). Precision also depended on context: *"For monitoring someone's health, you would have to be very accurate"* (P9)

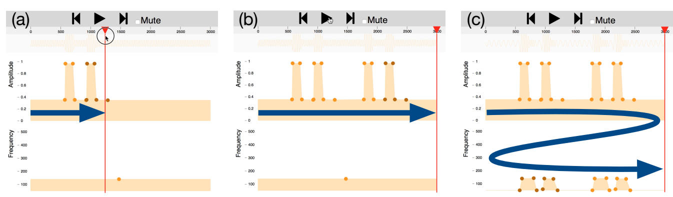
Editing and playback – During iteration, participants edited in bursts of primarily scrubbing activity, bookended by full playthroughs. They took time to realize each new version of the design before observing an overview. When editing, participants scrubbed back-and-forth, varying speed (P1-4,7,9,10), and dragging keyframes to try ideas out (P1,3,4,7,9,10) Figure 5.7. This feature was valued



(a) P9's cat/`none` design progressed sequentially in time. Note the red playhead helping alignment in (b).



(b) P6's car/`lo` design progressed by component, developing the component then repeating it.



(c) P10's heartbeat/`vis` design progressed by track. Amplitude was developed first, then frequency.

Figure 5.6: Participants created their designs using different progression paths, suggesting flexibility.

by those who used it: *"The real-time part is pretty important"* (P1); some rarely played, showing more frequent or longer periods of scrubbing instead (P2,9,10). Others rarely scrubbed (P5, P8), possibly to have an overall sense of the design: *"Trying to get a general sense of how it might feel"* (P8). P3, P4, and P7 all exhibited focused editing with mute enabled, unmuting for the bookended play sections; others did not use muting.

Encoding and Framing – Some participants encoded parameters using consistent rules, often aligned to events like heartbeats or lightning bolts. Others sought to create moods or metaphors for sensation.

Encoding was most visible in the lightning task, where participants represented lightning bolts in regular ways: *"if there was a lightning bolt on the left, I put amplitude and frequency a little longer than a lightning bolt on the right"* (P9). When the animation had two simultaneous bolts, several (P2-4,7,9) encoded it by

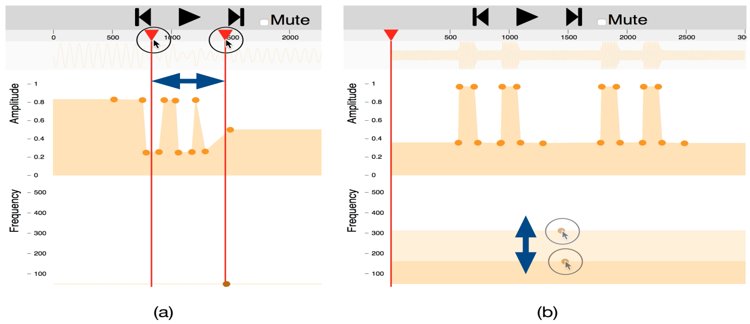


Figure 5.7: Participants used real-time feedback to explore, both (a) in time by scrubbing back and forth (P3 lightning/`lo`), and (b) by moving keyframes (P10 heartbeat/`vis`).

superimposing two bolt representations on top of one another. Participants were forced to reframe their encoding strategy: *"...two [lightning bolts]...I divided it into two equal partitions, .6 and 1"* (P7).

Encoding failed when participants did not find a direct mapping: *"When the three [snow flakes] come together I think my strategy broke down"* (P7). Metaphors helped in these cases. Car took extra imagination, either for the experience of driving (P6, P8, and P9 didn't drive), or because it's hard to *"know what it would feel like on the wrist"* (P1). P6 describes her process for both lightning and snow as using mood: *"...what I think the mood is...like snow fall, it's kinda like, very gentle and calm"* (P6).

5.6.3 Example Use

As seen, examples played a major role in users' design processes. Analysis revealed the effect of examples to be more nuanced than a one-to-one mapping of the theoretical dimensions of incorporability and visibility. Emergent themes were instead organized on the *role* of examples: as a *direct starting point* for each design; and to *indirectly scaffold learning* throughout a session. The latter was related to additional themes: task difficulty and individual differences.

Direct example use – task starting point –

When participants *prepared* for each task by browsing to find a best-match example, then using it as a starting point, they did this with a spectrum of strategies. These strategies, elaborated in Table 5.3, range from Ignore (examples not used) to Select (an example was the final design).

Indirect example use – observe how to design – Over the course of the session,

Ignore	Deliberately do not choose an example, through either lack of match: *"I didn't [find] the examples that I wanted"* (P1); a desire to challenge themselves or be creative: *"I wanted to do my own thing!"* (P9); or difficulty in using the examples.
Inspire	Choose an example, but do not explicitly copy/paste or replicate it in the editor; instead, design based loosely on example parts, sometimes as an adaptation from memory: *"I just tried to remember what the keyframes were like before, and then I modified it"* (P6 car/`lo`).
Template	Choose an initial example, but alter it considerably. In this case, participants use the example to expedite the process.
Adjust	Find an initial example, skipped major iteration and went directly to the refine stage, sometimes because the example was a close match. To enable this, some participants wanted a more powerful manipulation methods, like inverting (P7).
Select	Copy/paste an example (or manually recreate it), then do not modify; sometimes because the example seemed to match: *"...copy and paste, then confirmed it was the same."* (P5)

Table 5.3: Strategies used by participants to directly use examples as a starting point. Ignore and Inspire did not start with copy/paste; Template, Adjust, and Select did, with varying amounts of editing afterwards. When copy/paste was not available, manual re-creation was used as a stand-in.

participants used underlying structures of examples to understand how to design VT icons. This was most evident in the `none` or `lo` condition after participants were first exposed to examples: *"I sort of remembered"* (P4 car/`none`). Some explicitly described learning: *"It gave me a general idea of thinking in big shapes rather than little dots"* (P9 lightning/`vis`).

Most participants commented on the difficulty or ease of their task (P1-5, 7-9). Task difficulty was connected learning (*"It's easy...maybe it's more experience"* (P4 snow/`lo`)) and individual differences. Some people were motivated to learn, and challenge themselves; others were not.

Connections between these factors are complex and difficult to unravel with this data. We speculate on the utility of flow theory [49] as a useful lens to connect these issues, as it considers creativity, education, and the relationship between perceived challenge and perceived ability. We plan to use it to frame future exploration.

5.7 Discussion

We discuss implications for design, then limitations we hope to progress on with future work.

5.7.1 Implications for Design

Expose example structures for learning – When exposed to examples' underlying structure, participants are able to build their repertoire and learn VT design conventions like *"big shapes"* (P9). Such scaffolding is particularly crucial in an environment where experienced VT designers and training possibilities are rare. Whether through exploratory tool use or structured with online training programs, examples can expand the VT design practices available to novice designers.

Examples as templates – Participants typically copied an example first before iterating and customizing, suggesting a template model of modifiable source documents as a way to expose structure and reduce effort for designers.

Example Recommender – The time participants spent searching for the suitable examples suggests a recommender system could be very valuable. AI techniques might recommend examples similar (or dissimilar) to a source stimuli, as with previous tools in other sensory modalities [129] and VT visualization tools like VibViz [201].

Clarify example context – Participants often repeated gallery searches for each new animation; they needed to compare examples alongside the target graphic. In addition, though our examples were designed independently of our animation tasks, some participants showed confusion about whether they were supposed to match. Clarifying the context for each example, by presenting it either in connection to its original design goal or as a candidate for the participant's current goal, will help participants choose an example.

Hideable examples – Some participants wanted to be individualistic with their designs and actively disliked the most powerful `hi` condition, saying that the `none` condition was cleaner, or that while examples were helpful to learn, they felt "more creative" with fewer examples present. A hideable gallery, which can be opened when needed but kept hidden otherwise, could accommodate user preference. An intelligent gallery could even time example appearances or suggestions to occur at helpful design stages, e.g., by recognizing by activity patterns [60, 227].

Realtime "prefeel" then render – Macaron's real-time feedback supported exploration, with full play-throughs providing an overview or evaluation in-between editing sessions. In addition, P4, who was familiar with haptics, felt that the scrubbing synthesis was "muddy" relative to waveforms pre-rendered with audio tools – a common challenge, noted also by the researchers but deemed suitable for this study. While we hope this technology deficit inspires improved realtime rendering algorithms, it also suggests an explicit workflow compromise. Many video editing and compositing tools show a low-resolution previsualization in design mode; a clip is then fully rendered for playback. For tactile design, coarse, "prefeel" sensations would be synthesized for immediate feedback during a rough design stage, and a high-fidelity rendering generated for less frequent play-throughs. This could help computationally demanding, perceptually-based models or multi-actuator setups (e.g., tactile animation [196] as a prefeel for tactile brush [105]).

Tool flexibility – Macaron was used in very different ways depending on the participant. Some progressed by time, by track, by component, or a combination thereof. Some mirrored frequency and amplitude, using them together, while others used them to express different ideas. This suggests that tools should be flexible and accommodate different strategies; perhaps offering a choice to group by parameters (e.g., [196]) or work along parameter tracks (e.g., [215, 216]).

Alignment tools – Participants frequently used the playhead for alignment, finding locations in the video or aligning points between amplitude and frequency. Participants requested using modifier keys to align points (as in other editing tools), or a visualization of events in video. This suggests several features, providing ability to:

– Align comparison sensations from each modality - visual or audio sensation alongside VT.

– Place anchors for attaching a VT sensation (or keyframe within it) to a point in a target visual or audio sensation. This might be automatically assisted, e.g, with video analysis techniques to find scene changes.

– Automatically align keyframes to nearby keyframes, or use a modifier key to constrain or nudge keyframe movement.

Reuse – Copy/paste, especially from a template, speeded design and facilitated otherwise tedious approaches. Several participants made use of element repetition, which had to be re-done upon design re-framing. While copy/paste was helpful, more powerful repetition tools (e.g. looping, and "master templates", as in PowerPoint) would likely find use by many designers.

Automated Encoding – Some participants applied consistent rules in translating an animation to a tactile rendering – e.g., representing left/right lightning bolts differently in the lightning animation, or directly matching amplitude to up-down motion in the car animation. Some of these practices might be automated into generative rules. For example, video analysis could detect up/down motion for a visual object, and translate that automatically to a level for amplitude, similar to how motion trackers can track a moving object and link that to position of an animation; or, a designer might want to specify the mapping. More complex parameterizations could provide a useful tool for expert users, much like how `fmod` allows for parametrized audio in game design.

5.7.2 Limitations & Future work

Limitations in our study suggest future lines of inquiry: following up on additions study factor by deploying online.

Study factors – Our Latin square design allowed qualitative comparison of several gallery variants, but did not have the power for comparative statistical tests between the alternatives. Meanwhile, five design tasks presented in a uniform order did not permit systematic insights into other factors: learning, or task features such as abstractness and complexity. Flow was identified after-the-fact as an important framework for future analysis, but only after our study was designed and data was collected.

Our proposed example-usage dimensions of visibility and incorporability were a useful starting point, but did not line up well with the task processes that people actually used with Macaron. We did see behaviors that aligned well with *learning* and *design-starting* from examples, as well as hints of a more rich and nuanced view of what makes examples useful and in what way.

First, the examples-as-starting-point strategies actually used (Table 5.3) suggest that visibility and incorporability at minimum are not quite right and probably insufficient in dimensionality – there is a concept of edibility regardless of starting point; whereas incorporability could entail editing, but certainly requires an example as a start.

Additionally, observations (including details not reported due to space limits) suggest other factors that influence example use, e.g., *difficulty*, from task, interface and personal confidence and experience; and *task*, from task complexity and abstraction, user strategy, e.g. encoding and metaphor, and user confidence and experience. These hints are far from orthogonal, and will require further research, with focus turned to elements like task abstraction and user background, to disentangle and prioritize.

Online deployment – Triangulation will be helpful in studying factors like difficulty, task abstraction, and user background. In this study, Macaron was deployed and studied locally. We were able to validate the editor's design support and utility of its logging methods, and expose many interesting insights into natural end-user design practices.

Our next plan to answer these questions is to deploy Macaron at a larger scale: online, as a free-to-use design tool for the haptics community, with an initial study in haptics courses. This will allow research *in-situ* with larger, more quantitative, remote-based methods for data collection, triangulated with the less scalable qualitative methods used in-lab. Interaction logs, use statistics, and A/B tests will help us further develop Macaron as a tool for VT design and more generally as a lens for the haptic design process.

5.8 Conclusion

In this paper, we present initial findings from a vibrotactile (VT) design gallery, Macaron. This tool revealed insights both into how examples are used in VT design and implications for other VT design tools. Macaron was implemented using web tools, offering a unique opportunity to follow-up on the design process we observed here, helping designers to create engaging experiences while understanding their craft.

5.9 Acknowledgements

We are grateful to our participants, labmates, and reviewers for all feedback. This work was supported by NSERC.

Chapter 6

Share: HapTurk

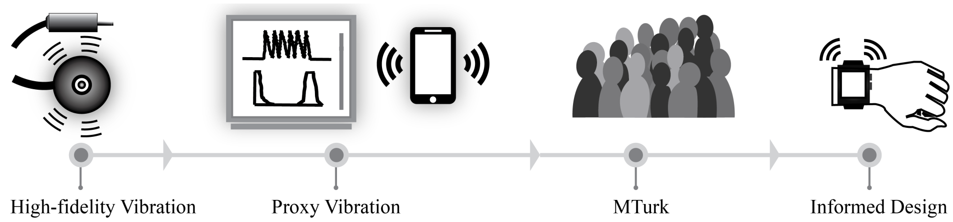


Figure 6.1: In HapTurk, we access large-scale feedback on informational effectiveness of high-fidelity vibrations after translating them into proxies of various modalities, rendering important characteristics in a crowdsource-friendly way.

Preface – While Chapters 3-5 describe iterative development of vibrotactile tools, HapTurk[1] studies a vibrotactile technique. HapTurk follows up on our goal of collaboration in Chapter 3, where we found utility in informal, collocated user feedback. Here, we look into *browse*'s opposite: *share*, disseminating or storing a design concept for others' use. In other design domains, crowdsourcing platforms like Amazon's MTurk can deploy user studies with large samples, receiving extremely rapid feedback. However, high-fidelity haptic sensations require a specialized device usually constrained to the lab. Our approach is to instead send more easily-shared stimuli: proxies, which are sent to the crowd instead of the source stimuli. Though we use proxies to collect crowdsourced feedback, this technique be used for other *sharing* purposes, *e.g.*, media broadcasting.

[1]Schneider, Seifi, Kashani, Chun, and MacLean. (2016) *HapTurk: Crowdsourcing Affective Ratings for Vibrotactile Icons*. Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16.

6.1 Abstract

Vibrotactile (VT) display is becoming a standard component of informative user experience, where notifications and feedback must convey information eyes-free. However, effective design is hindered by incomplete understanding of relevant perceptual qualities. To access evaluation streamlining now common in visual design, we introduce *proxy modalities* as a way to crowdsource VT sensations by reliably communicating high-level features through a crowd-accessible channel. We investigate two proxy modalities to represent a high-fidelity tactor: a new VT visualization, and low-fidelity vibratory translations playable on commodity smartphones. We translated 10 high-fidelity vibrations into both modalities, and in two user studies found that both proxy modalities can communicate affective features, and are consistent when deployed remotely over Mechanical Turk. We analyze fit of features to modalities, and suggest future improvements.

6.2 Introduction

In modern handheld and wearable devices, vibrotactile (VT) feedback can provide unintrusive, potentially meaningful cues through wearables in on-the-go contexts [24]. With consumer wearables like Pebble and the Apple Watch featuring high-fidelity actuators, VT feedback is becoming standard in more user tools. Today, VT designers seek to provide sensations with various perceptual and emotional connotations to support the growing use cases for VT feedback (everyday apps, games, etc.). Although low-level design guidelines exist and are helpful for addressing perceptual requirements [17, 23, 102, 221**?**], higher-level concerns and design approaches to increase their usability and information capacity (e.g., a user's desired affective response, or affective or metaphorical interpretation) have only recently received study and are far from solved [5, 108, 111, 162, 164, 200]. Tactile design thus relies heavily on iteration and user feedback [191]. Despite its importance [200, 201], collecting user feedback on perceptual and emotional (i.e., affective) properties of tactile sensations in small-scale lab studies is undermined by noise due to individual differences (IDs).

In other design domains, crowdsourcing enables collecting feedback at scale. Researchers and designers use platforms like Amazon's Mechanical Turk (`www.mturk.com`) to deploy user studies with large samples, receiving extremely rapid feedback in, e.g., creative text production [**?**], graphic design [**?**] and sonic imitations [**?**].

The problem with crowdsourcing tactile feedback is that the "crowd" can't feel the stimuli. Even when consumer devices have tactors, output quality and intensity is unpredictable and uncontrollable. Sending each user a device is impractical.

What we need are crowd-friendly proxies for test stimuli. Here, we define a *proxy vibration* as a sensation that communicates key characteristics of a source stimulus within a bounded error; a *proxy modality* is the perceptual channel and representation employed. In the new evaluation process thus enabled, the designer translates a sensation of interest into a proxy modality, receives rapid feedback from a crowd-sourcing platform, then interprets that feedback using known error bounds. In this way, designers can receive high-volume, rapid feedback to use in tandem with costly in-lab studies, for example, to guide initial designs or to generalize findings from smaller studies with a larger sample.

To this end, we must first establish feasibility of this approach, with specific goals: **(G1)** Do proxy modalities work? Can they effectively communicate both physical VT properties (e.g., duration), and high-level affective properties (roughness, pleasantness)? **(G2)** Can proxies be deployed remotely? **(G3)** What modalities work, and **(G4)** what obstacles must be overcome to make this approach practical?

This paper describes a proof-of-concept for proxy modalities for tactile crowdsourcing, and identifies challenges throughout the workflow pipeline. We describe and assess two modalities' development, translation process, validation with a test set translation, and MTurk deployment. Our two modalities are a new technique to graphically visualize high-level traits, and the low-fidelity actuators on users' own commodity smartphones. Our test material is a set of 10 VT stimuli designed for a high-fidelity tactile display suitable for wearables (referred to as "high fidelity vibrations"), and perceptually well understood as presented by that type of display (Figure 6.7). We conducted two coupled studies, first validating proxy expressiveness in lab, then establishing correspondence of results in remote deployment. Our contributions are:

- A way to crowdsource tactile sensations (vibration proxies), with a technical proof-of-concept.
- A visualization method that communicates high-level affective features more effectively than the current tactile visualization standard (vibration waveforms).
- Evidence that both proxy modalities can represent high-level affective features, with lessons about which features work best with which modalities.
- Evidence that our proxy modalities are consistently rated in-lab and remotely, with initial lessons for compliance.

6.3 Related Work

We cover work related to VT icons and evaluation methods for VT effects, the current understanding of affective haptics, and work with Mechanical Turk in other modalities.

6.3.1 Existing Evaluation Methods for VT Effects

The haptic community has appropriated or developed many types of user studies to evaluate VT effects and support VT design. These target a variety of objectives:

1) *Perceptibility:* Determine the perceptual threshold or Just Noticeable Difference (JND) of VT parameters. Researchers vary the values of a VT parameter (e.g., frequency) to determine the minimum perceptible change [**? ?**].

2) *Illusions:* Studies investigate effects like masking or apparent motion of VT sensations, useful to expand a haptic designer's palette [93, 105, 203].

3) *Perceptual organization:* Reveal the underlying dimensionality of how humans perceive VT effects (which are generally different than the machine parameters used to generate the stimuli). Multidimensional Scaling (MDS) studies are common, inviting participants compare or group vibrations based on perceived similarity [35, 221**? ? ?**].

4) *Encoding abstract information:* Researchers examine salient and memorable VT parameters (e.g. energy, rhythm) as well as the number of VT icons that people can remember and attribute to an information piece [23, 35, 221**?**].

5) *Assign affect:* Studies investigate the link between affective characteristics of vibrations (e.g., pleasantness, urgency) to their engineering parameters (e.g., frequency, waveform) [221**? ? ?**]. To achieve this, VT researchers commonly design or collect a set of vibrations and ask participants to rate them on a set of qualitative metrics.

6) *Identify language:* Participants describe or annotate tactile stimuli in natural language [35, 85, 162, 201, 221**?**].

7) *Use case support:* Case studies focus on conveying information with VT icons such as collaboration [35], public transit [24] and direction [5, 24], or timing of a presentation [**?**]. In other cases, VT effects are designed for user engagement, for example in games and movies, multimodal storytelling, or art installations [108**?**]. Here, the designers use iterative design and user feedback (qualitative and quantitative with user rating) to refine and ensure effective design.

All of the above studies would benefit from the large number of participants and fast data collection on MTurk. In this paper, we chose our methodology so that the results are informative for a broad range of these studies.

6.3.2 Affective Haptics

VT designers have the challenge of creating perceptually salient icon sets that convey meaningful content. A full range of expressiveness means manipulating not only a vibration's physical characteristics but also its perceptual and emotional properties, and collecting feedback on this. Here, we refer to all these properties as affective characteristics.

Some foundations for affective VT design are in place. Studies on tactile language and affect are establishing a set of perceptual metrics [162, 201]. Guest *et al* collated a large list of emotion and sensation words describing tactile stimuli; then, based on multidimensional scaling of similarity ratings, proposed comfort or pleasantness and arousal as key dimensions for tactile emotion words, and rough/smooth, cold/warm, and wet/dry for sensation [162]. Even so, there is not yet agreement on an affective tactile design language [111].

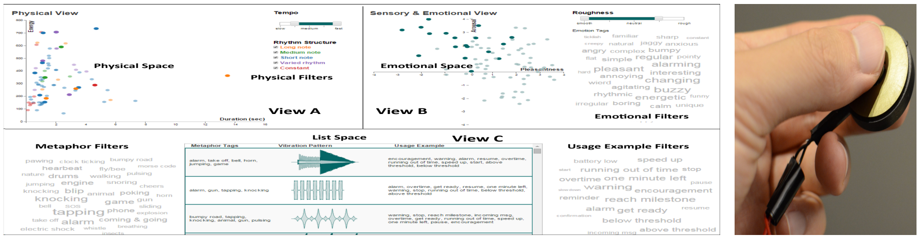
Recently, Seifi *et al* compiled research on tactile language into five taxonomies for describing vibrations [201]. **1) Physical properties** that can be measured: e.g., duration, energy, tempo or speed, rhythm structure; **2) sensory properties**: roughness, and sensory words from Guest *et al*'s touch dictionary [85]; **3) emotional interpretations**: pleasantness, arousal (urgency), dictionary emotion words [85]; **4) metaphors** provide familiar examples resembling the vibration's feel: heartbeat, insects; **5) usage examples** describe events which a vibration fits: an incoming message or alarm.

To evaluate our vibration proxies, we derived six metrics from these taxonomies to capture vibrations' physical, sensory and emotional aspects: 1) duration, 2) energy, 3) speed, 4) roughness, 5) pleasantness, and 6) urgency.

6.3.3 Mechanical Turk (MTurk)

MTurk is a platform for receiving feedback from a large number of users, in a short time at a low cost [**?** **?**]. These large, fast, cheap samples have proved useful for many cases including running perceptual studies [**?**], developing taxonomies [**?**], feedback on text [**?**], graphic design [**?**], and sonic imitations [**?**].

Crowdsourced studies have drawbacks. The remote, asynchronous study environment is not controlled; compared to a quiet lab, participants may be subjected to unknown interruptions, and may spend less time on task with more response variability [**?**]. MTurk is not suitable for getting rich, qualitative feedback or following up on performance or strategy [**?**]. Best practices – e.g., simplifying tasks to be confined to a singular activity, or using instructions complemented with example responses – are used to reduce task ambiguity and improve response quality [**?**]. Some participants try to exploit the service for personal profit, exhibiting low task



(a) VibViz interface [201] **(b)** C2 tactor

Figure 6.2: Source of high-fidelity vibrations and perceptual rating scales.

engagement [**?**], and must be pre- or post-screened.

Studies have examined MTurk result validity in other domains. Most relevantly, Heer *et al* [**?**] validated MTurk data for graphical perception experiments (spatial encoding and luminance contrast) by replicating previous perceptual studies on MTurk. Similarly, we compare results of our local user study with an MTurk study to assess viability of running VT studies on MTurk, and collect and examine phone properties in our MTurk deployment.

Need for HapTurk: Our present goal is to give the haptic design community access to crowdsourced evaluation so we can establish modality-specific methodological tradeoffs. There is ample need for huge-sample haptic evaluation. User experience of transmitted sensations must be robust to receiving device diversity. Techniques to broadcast haptic effects to video [123, 153], e.g., with YouTube [1] or MPEG7 [61, 62] now require known high-fidelity devices because of remote device uncertainty; the same applies to social protocols developed for remote use of high-quality vibrations, e.g. in collaborative turn taking [35]. Elsewhere, studies of VT use in consumer devices need larger samples: e.g., perceivability [**?**], encoding of caller parameters [**?**], including caller emotion and physical presence collected from pressure on another handset [98], and usability of expressive, customizable VT icons in social messaging [109]. To our knowledge, this is the first attempt to run a haptic study on a crowdsource site and characterize its feasibility and challenges for haptics.

6.4 Sourcing reference vibrations and qualities

We required a set of exemplar source vibrations on which to base our proxy modalities. This set needed to 1) vary in physical, perceptual, and emotional characteristics, 2) represent the variation in a larger source library, and 3) be small enough

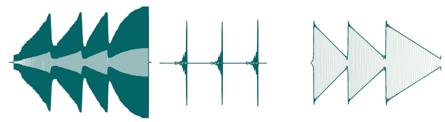


Figure 6.3: VIS_{DIR} Visualization, based on VibViz

for experimental feasibility.

6.4.1 High-fidelity reference library

We chose 10 vibrations from a large, freely available library of 120 vibrations (VibViz, [201]), browsable through five descriptive taxonomies, and ratings of taxonomic properties. Vibrations were designed for an Engineering Acoustics C2 tactor, a high-fidelity, wearable-suitable voice coil, commonly used in haptic research [201]. We employed VibViz's filtering tools to sample, ensuring variety and coverage by selecting vibrations at high and low ends of energy / duration dimensions, and filtering by ratings of temporal structure/rhythm, roughness, pleasantness, and urgency. To reduce bias, two researchers independently and iteratively selected a set of 10 items each, which were then merged.

Because VibViz was designed for a C2 tactor, we used a handheld C2 in the present study (Figure 6.2b).

6.4.2 Affective properties and rating scales

To evaluate our proxies, we adapted six rating scales from the tactile literature and new studies. Seifi *et al* [201] proposed five taxonomies for describing vibrations including physical, sensory, emotional, metaphors, and use examples. Three taxonomies comprise quantitative metrics and adjectives; two use descriptive words.

We chose six quantitative metrics from [201] that capture important affective (physical, perceptual, and emotional) VT qualities: 1) *duration* [low-high], 2) *energy* [low-high], 3) *speed* [slow-fast], 4) *roughness* [smooth-rough], 5) *urgency* [relaxed-alarming], and 6) *pleasantness* [unpleasant-pleasant]. A large scale (0-100) allowed us to treat the ratings as continuous variables. To keep trials quick and MTurk-suitable, we did not request open-ended responses or tagging.

6.5 Proxy Choice and Design

The proxies' purpose was to capture high-level traits of source signals. We investigated two proxy channels and approaches, to efficiently establish viability and search for triangulated perspectives on what will work. The most obvious

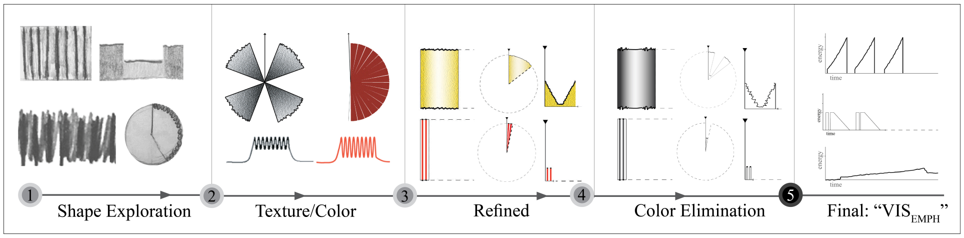


Figure 6.4: Visualization design process. Iterative development and piloting results in the VIS$_{\text{EMPH}}$ visualization pattern.

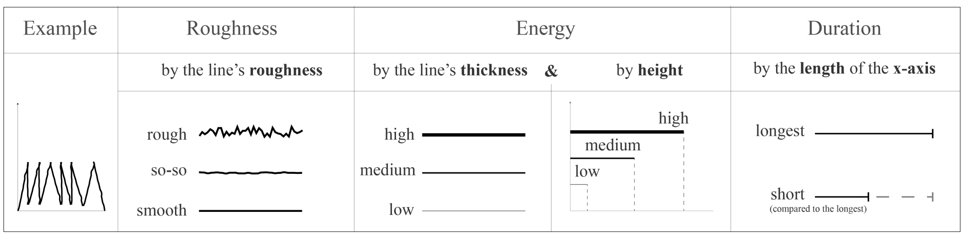


Figure 6.5: Final VIS$_{\text{EMPH}}$ visualization guide, used by researchers to create VIS$_{\text{EMPH}}$ proxy vibrations and provided to participants during VIS$_{\text{EMPH}}$ study conditions.

starting points are to 1) visually augment the current standard of a direct trace of $amplitude = f(time)$, and 2) reconstruct vibrations for common-denominator, low-fidelity actuators.

We considered other possibilities (e.g., auditory stimuli, for which MTurk has been used [**?**], or animations). However, our selected modalities balance a) directness of translation (low fidelity could not be excluded); b) signal control (hard to ensure consistent audio quality/volume/ambient masking); and c) development progression (visualization underlies animation, and is simpler to design, implement, display). We avoided multisensory combinations at this early stage for clarity of results. Once the key modalities are tested, combinations can be investigated in future work.

"REF" denotes high-fidelity source renderings (C2 tactor).

1) Visual proxies: Norms in published works (e.g. [35]) directed [201] to confirm that users rely on graphical $f(time)$ plots to skim and choose from large libraries. We tested the direct plot, VIS$_{\text{DIR}}$, as the status-quo representation.

However, these unmodified time-series emphasize or mask traits differently than felt vibrations, in particular for higher-level or "meta" responses. We considered many other means of visualizing vibration characteristics, pruned candidates and refined design via piloting to produce a new scheme which explicitly *emphasizes* affective features, VIS_{EMPH}.

2) Low-fidelity vibration proxy: Commodity device (e.g. smartphone) actuators usually have low output capability compared to the C2, in terms of frequency response, loudness range, distortion and parameter independence. Encouraged by expressive rendering of VT sensations with commodity actuation (from early constraints [35] to deliberate design-for-lofi [109]), we altered stimuli to convey high-level parameters under these conditions, hereafter referred to as LOFIVIB.

Translation: Below, we detail first-pass proxy development. In this feasibility stage, we translated proxy vibrations manually and iteratively, as we sought generalizable mappings of the parametric vibration definition to the perceptual quality we wished to highlight in the proxy. We frequently relied on a cycle of user feedback, e.g., to establish the perceived roughness of the original stimuli and proxy candidate.

Automatic translation is an exciting goal. Without it, HapTurk is still useful for gathering large samples; but automation will enable a very rapid create-test cycle. It should be attainable, bootstrapped by the up-scaling of crowdsourcing itself. With a basic process in place, we can use MTurk studies to identify these mappings relatively quickly.

6.5.1 Visualization Design (VIS_{DIR} and VIS_{EMPH})

VIS_{DIR} was based on the original waveform visualization used in VibViz (Figure 6.3). In Matlab, vibration frequency and envelope were encoded to highlight its pattern over time. Since VIS_{DIR} patterns were detailed, technical and often inscrutable for users without an engineering background, we also developed a more interpretive visual representation, VIS_{EMPH}; and included VIS_{DIR} as a status-quo baseline.

We took many approaches to depicting vibration high-level properties, with visual elements such as line thickness, shape, texture and colour (Figure 6.4). We first focused on line sharpness, colour intensity, length and texture: graphical waveform smoothness and roughness were mapped to perceived roughness; colour intensity highlighted perceived energy. Duration mapped to length of the graphic, while colour and texture encoded the original's invoked emotion.

Four participants were informally interviewed and asked to feel REF vibrations, describe their reactions, and compare them to several visualization candidates. Participants differed in their responses, and had difficulties in understanding VT emotional characteristics from the graphic (i.e. pleasantness, urgency), and in reading the circular patterns. We simplified the designs, eliminating representation of emotional characteristics (color, texture), while retaining more objective mappings for physical and sensory characteristics.

VIS$_{\text{EMPH}}$ won an informal evaluation of final proxy candidates (n=7), and was captured in a translation guideline (Figure 6.5).

6.5.2 Low Fidelity Vibration Design

For our second proxy modality, we translated REF vibrations into LOFIVIB vibrations. We used a smartphone platform for their built-in commodity-level VT displays, their ubiquity amongst users, and low security concerns for vibration imports to personal devices [**?**]. To distribute vibrations remotely, we used HTML5 Vibration API, implemented on Android phones running compatible web browsers (Google Chrome or Mozilla Firefox).

As with VIS$_{\text{EMPH}}$, we focused on physical properties when developing LOFIVIB(our single low-fi proxy exemplar). We emphasized rhythm structure, an important design parameter [221] and the only direct control parameter of the HTML5 API, which issues vibrations using a series of on/off durations. Simultaneously, we manipulated perceived energy level by adjusting the actuator pulse train on/off ratio, up to the point where the rhythm presentation was compromised. Shorter durations represented a weak-feeling hi-fi signal, while longer durations conveyed intensity in the original. This was most challenging for dynamic intensities or frequencies, such as increasing or decreasing ramps, and long, low-intensity sensations. Here we used a duty-cycle inspired technique, similar to [109], illustrated in Figure 6.6.

To mitigate the effect of different actuators found in smartphones, we limited our investigation to Android OS. While this restricted our participant pool, there was nevertheless no difficulty in quickly collecting data for either study. We designed for two phones representing the largest classes of smartphone actuators: Samsung Galaxy Nexus, which contains a coin-style actuator, and a Sony Xperia Z3 Compact, which uses a pager motor resulting in more subdued, smooth sensations. Though perceptually different, control of both actuator styles are limited to on/off durations. As with VIS$_{\text{EMPH}}$, we developed LOFIVIB vibrations iteratively, first with team feedback, then informal interviews (n=6).

6.6 Study 1: In-lab Proxy Vibration Validation (G1)

We obtained user ratings for the hi-fi source vibrations REFand three proxies (VIS$_{\text{DIR}}$, VIS$_{\text{EMPH}}$, and LOFIVIB). An in-lab format avoided confounds and unknowns due to remote MTurk deployment, addressed in Study 2. Study 1 had two versions: in

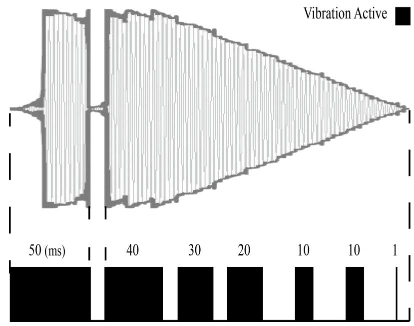


Figure 6.6: Example of LOFIVIB proxy design. Pulse duration was hand-tuned to represent length and intensity, using duty cycle to express dynamics such as ramps and oscillations.

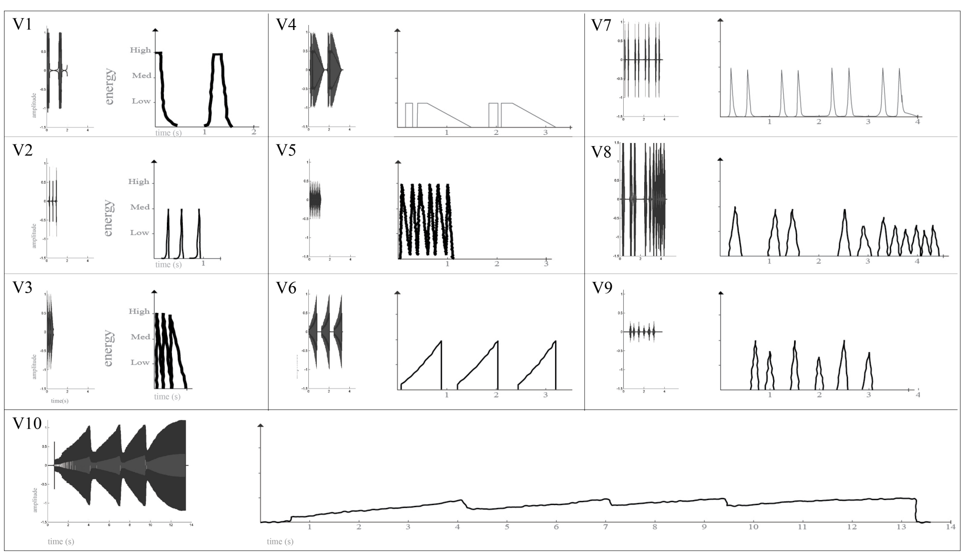


Figure 6.7: Vibrations visualized as both VIS_{DIR} (left of each pair) and VIS_{EMPH}.

one, participants rated visual proxies VIS_{DIR} and VIS_{EMPH} next to REF; and in the other, LOFIVIB next to REF. REF_{VIS} and $\text{REF}_{\text{LOFIVIB}}$ denote these two references, each compared with its respective proxy(ies) and thus with its own data. In each substudy, participants rated each REF vibration on 6 scales [0-100] in a computer survey, and again for the proxies. Participants in the visual substudy did this for both VIS_{DIR} and VIS_{EMPH}, then indicated preference for one. Participants in the lo-fi study completed the LOFIVIB survey on a phone, which also played vibrations using Javascript and HTML5; other survey elements employed a laptop. 40 participants aged 18-50 were recruited via university undergraduate mailing lists. 20 (8F) participated in the visual substudy, and a different 20 (10F) in the low-fi vibration substudy.

Reference and proxies were presented in different random orders. Pilots confirmed that participants did not notice proxy/target linkages, and thus were unlikely to consciously match their ratings between pair elements. REF/proxy presentation order was counterbalanced, as was $\text{VIS}_{\text{DIR}}/\text{VIS}_{\text{EMPH}}$.

6.6.1 Comparison Metric: Equivalence Threshold

To assess whether a proxy modalities were rated similarly to their targets, we employed *equivalence testing*, which tests the hypothesis that sample means are within a threshold δ, against the null of being outside it [**?**]. This tests if two samples are equivalent with a known error bound; it corresponds to creating confidence intervals of means, and examining whether they lie entirely within the range $(-\delta, \delta)$.

We first computed least-squares means for the 6 rating scales for each proxy modality and vibration. 95% confidence intervals (CI) for REF rating means ranged from 14.23 points (Duration ratings) to 20.33 (Speed). Because estimates of the REF "gold standard" mean could not be more precise than these bounds, we set equivalence thresholds for each rating equal to CI width. For example, given the CI for Duration of 14.23, we considered proxy Duration ratings equivalent if the CI for a difference fell completely in the range $(-14.23, 14.23)$. With pooled standard error, this corresponded to the case where two CIs overlap by more than 50%. We also report when a *difference* was detected, through typical hypothesis testing (i.e., where CIs do not overlap).

Thus, each rating set pair could be *equivalent*, *uncertain*, or *different*. Figure 6.9 offers insight into how these levels are reflected in the data given the high rating variance. This approach gives a useful error bound, quantifying the precision tradeoff in using vibration proxies to crowdsource feedback.

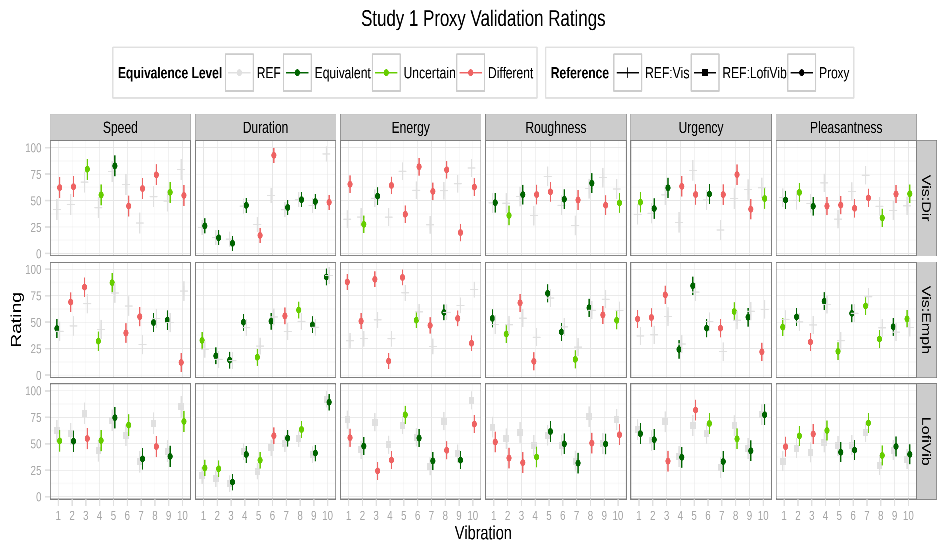


Figure 6.8: 95% confidence intervals and equivalence test results for Study 1 - Proxy Validation. Grey represents REF ratings. Dark green maps equivalence within our defined threshold, and red a statistical difference indicating an introduced bias; light green results are inconclusive. Within each cell, variation of REF ratings means vibrations were rated differently compared to each other, suggesting they have different perceptual features and represent a varied set of source stimuli.

6.6.2 Proxy Validation (Study 1) Results and Discussion

Overview of Results

Study 1 results appear graphically in Figure 6.8. To interpret this plot, look for (1) equivalence indicated by bar color, and CI size by bar height (dark green/small are good); (2) rating richness: how much spread, vibration to vibration, within a cell indicates how well that parameter captures the differences users perceived; (3) modality consistency: the degree to which the bars' up/down pattern translates vertically across rows. When similar (and not flat), the proxy translations are being interpreted by users in the same way, providing another level of validation. We structure our discussion around how the three modalities represent the different rating scales. We refer to the number of *equivalents* and *differents* in a given cell as $[x{:}z]$, with $y =$ number of *uncertains*, and $x+y+z = 10$.

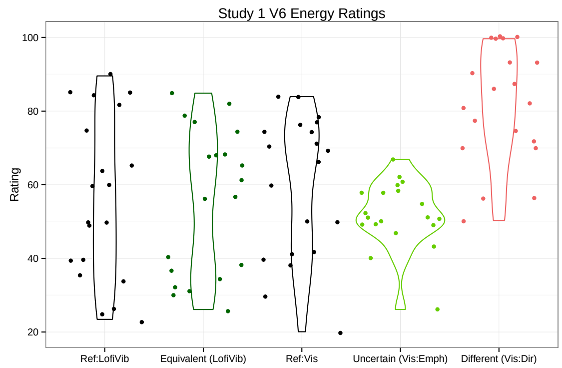


Figure 6.9: Rating distributions from Study 1, using V6 Energy as an example. These violin plots illustrate 1) the large variance in participant ratings, and 2) how equivalence thresholds reflect the data. When equivalent, proxy ratings are visibly similar to REF. When uncertain, ratings follow a distribution with unclear differences. When different, there is a clear shift.

Duration and Pleasantness were translatable

Duration was comparably translatable for LOFIVIB [5:1] and VIS_{EMPH} [6:1]; VIS_{DIR} was less consistent [7:3] (two differences very large). Between the three modalities, 9/10 vibrations achieved equivalence with at least one modality. For Duration, this is unsurprising. It is a physical property that is controllable through the Android vibration API, and both visualization methods explicitly present Duration as their *x*-axis. This information was apparently not lost in translation.

More surprisingly, Pleasantness fared only slightly worse for LOFIVIB [4:2] and VIS_{EMPH} [4:1]; 8 / 10 vibrations had at least one modality that provided equivalence. Pleasantness is a higher-level affective feature than Duration. Although not an absolute victory, this result gives evidence that, with improvement, crowdsourcing may be a viable method of feedback for at least one affective parameter.

Speed and Urgency translated better with LOFIVIB

LOFIVIB was effective at representing Urgency [6:2]; VIS_{EMPH} attained only [4:5], and VIS_{DIR} [3:5]. Speed was less translatable. LOFIVIB did best at [4:2]; VIS_{DIR} reached only [1:6], and VIS_{EMPH} [3:5]. However, the modalities again complemented each other. Of the three, 9/10 vibrations were equivalent at least once for Urgency (V8 was not). Speed had less coverage: 6/10 had equivalencies

(V3,4,6,10 did not).

Roughness had mixed results; best with VIS$_{\text{EMPH}}$

Roughness ratings varied heavily by vibration. 7 vibrations had at least one equivalence (V2,4,10 did not). All modalities had 4 equivalencies each: VIS$_{\text{EMPH}}$ [4:3], VIS$_{\text{DIR}}$ [4:4], and LOFIVIB [4:5].

Energy was most challenging

Like Roughness, 7 vibrations had at least one equivalence between modalities (V1,4,10 did not). LOFIVIB [4:5] did best with Energy; VIS$_{\text{EMPH}}$ and VIS$_{\text{DIR}}$ struggled at [1:8].

Emphasized visualization outperformed direct plot

Though it depended on the vibration, VIS$_{\text{EMPH}}$ outperformed VIS$_{\text{DIR}}$ for most metrics, having the same or better equivalencies/differences for Speed, Energy, Roughness, Urgency, and Pleasantness. Duration was the only mixed result, as VIS$_{\text{DIR}}$ had both more equivalencies and more differences [7:3] versus [6:1] In addition, 16/20 participants (80%) preferred VIS$_{\text{EMPH}}$ to VIS$_{\text{DIR}}$. Although not always clear-cut, these comparisons overall indicate that our VIS$_{\text{EMPH}}$ visualization method communicated these affective qualities more effectively than the status quo. This supports our approach to emphasized visualization, and motivates the future pursuit of other visualizations.

V4,V10 difficult, V9 easy to translate

While most vibrations had at least one equivalency for 5 rating scales, V4 and V10 only had 3. V4 and V10 had no equivalences at all for Speed, Roughness, and Energy, making them some of the most difficult vibrations to translate. V4's visualization had very straight lines, perhaps downplaying its texture. V10 was by far the longest vibration, at 13.5s (next longest was V8 with 4.4s). Its length may have similarly masked textural features.

V8 was not found to be equivalent for Urgency and Pleasantness. V8 is an extremely irregular vibration, with a varied rhythm and amplitude, and the second longest. This may have made it difficult to glean more intentional qualities like Urgency and Pleasantness. However, it was only found to be different for VIS$_{\text{DIR}}$/Urgency, so we cannot conclude that significant biases exist.

By contrast, V9 was the only vibration that had an equivalency for every rating scale, and in fact could be represented across all ratings with LOFIVIB. V9 was a

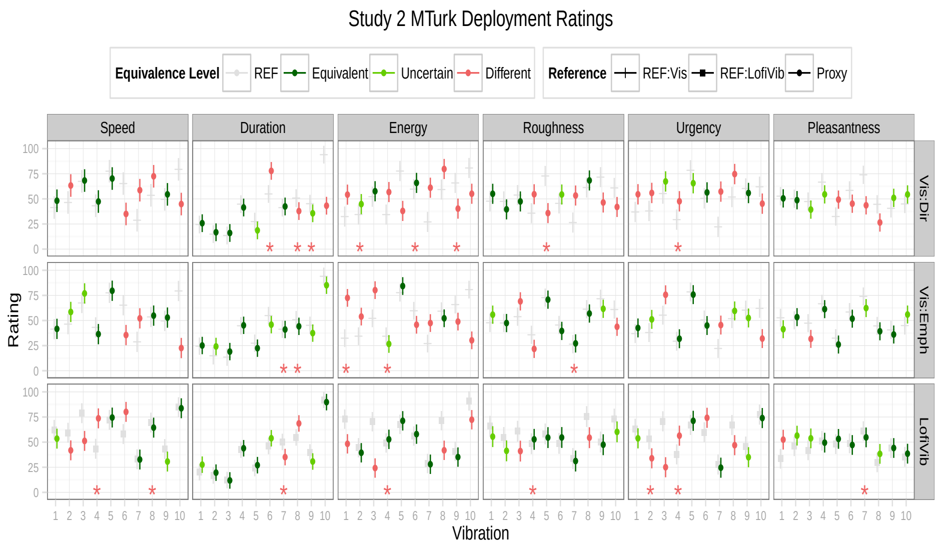


Figure 6.10: 95% Confidence Intervals and Equivalence Test Results for Study 2 - MTurk Deployment Validation. Equivalence is indicated with dark green, difference is indicated with red, and uncertainty with light green. Red star indicates statistically significant difference between remote and local proxy ratings.

set of distinct pulses, with no dynamic ramps; it thus may have been well suited to translation to LOFIVIB.

Summary

In general, these results indicate promise, but also need improvement and combination of proxy modalities. Unsurprisingly, participant ratings varied, reducing confidence and increasing the width of confidence intervals (indeed, this is partial motivation to access larger samples). Even so, both differences and equivalencies were found in every rating/proxy modality pairing. Most vibrations were equivalent with at least one modality, suggesting that we might pick an appropriate proxy modality depending on the vibration; we discuss the idea of triangulation in more detail later. Duration and Pleasantness were fairly well represented, Urgency and Speed were captured best by LOFIVIB, and Roughness was mixed. Energy was particularly difficult to represent with these modalities. We also find that results varied depending on vibration, meaning that more analysis into what makes vibrations easier or more difficult to represent could be helpful.

Though we were able to represent several features using proxy modalities within a bounded error rate, this alone does not mean they are crowdsource-friendly. All results from Study 1 were gathered in-lab, a more controlled environment than over MTurk. We thus ran a second study to validate our proxy modality ratings when deployed remotely.

6.7 Study 2: Deployment Validation with MTurk (G2)

To determine whether rating of a proxy is similar when gathered locally or remotely, we deployed the same computer-run proxy modality surveys on MTurk. We wanted to discover the challenges all through the pipeline for running a VT study on MTurk, including larger variations in phone actuators and experimental conditions (G4). We purposefully did not iterate on our proxy vibrations or survey, despite identifying many ways to improve them, to avoid creating a confound in comparing results of the two studies.

The visualization proxies were run as a single MTurk Human Intelligence Task (HIT), counterbalanced for order; the LOFIVIB survey was deployed as its own HIT. Each HIT was estimated at 30m, for which participants received $2.25 USD. In comparison, Study 1 participants were estimated to take 1 hour and received $10 CAD. We anticipated a discrepancy in average task time due to a lack of direct supervision for the MTurk participants, and expected this to lead to less accurate participant responses, prompting the lower payrate. On average, it took 7m for participants to complete the HIT while local study participants took 30m.

We initially accepted participants of any HIT approval rate to maximize recruitment in a short timeframe. Participants were post-screened to prevent participation in both studies. 49 participants were recruited. No post-screening was used for the visual sub-study. For the LOFIVIB proxy survey, we post-screened to verify device used [**?**]. We asked participants (a) confirm their study completion with an Android device via a survey question (b) detected actual device via FluidSurvey's OS-check feature, and (c) rejected inconsistent samples (eg. 9 used non-Android platforms for LOFIVIB). Of the included data, 20 participants participated each in the visual proxy condition (6F) and the LOFIVIB condition (9F).

For both studies, Study 1's data was used as a "gold standard" that served as a baseline comparison with the more reliable local participant ratings [**?**]. We compared the remote proxy results (from MTurk) to the REF results gathered in Study 1, using the same analysis methods.

6.7.1 Results

Study 2 results appear in Figure 6.10, which compares remotely collected ratings with locally collected ratings for the respective reference (the same reference as for Figure 6.8). It can be read the same way, but adds information. Based an analysis of a different comparison, a red star indicates a statistically significant difference between remote proxy ratings and corresponding local *proxy* ratings. This analysis revealed that ratings for the same proxy gathered remotely and locally disagreed 21 times (stars) out of 180 rating/modality/vibration combination; i.e., relatively infrequently.

Overall, we found similar results and patterns in Study 2 as for Study 1. The two figures show similar up/down rating patterns; the occasional exceptions correspond to red-starred items. Specific results varied, possibly due to statistical noise and rating variance. We draw similar conclusions: that proxy modalities can still be viable when deployed on MTurk, but require further development to be reliable in some cases.

6.8 Discussion

Here we discuss high level implications from our findings and relate them to our study goals (G1-G4 in Introduction).

6.8.1 Proxy modalities are viable for crowdsourcing (G1,G2:feasibility)

Our studies showed that proxy modalities can represent affective qualities of vibrations within reasonably chosen error bounds, depending on the vibration. These results largely translate to deployment on MTurk. Together, these two steps indicate that proxy modalities are be a viable approach to crowdsourcing VT sensations, and can reach a usable state with a bounded design iteration (as outlined in the following sections). This evidence also suggests that we may be able to deploy directly to MTurk for future validation. Our two-step validation was important as a first look at whether ratings shift dramatically; and we saw no indications of bias or overall shift between locally running proxy modalities and remotely deploying them.

6.8.2 Triangulation (G3:promising directions/proxies)

Most vibrations received equivalent ratings for most scales in at least one proxy modality. Using proxy modalities in tandem might help improve response accuracy. For example, V6 could be rendered with LOFIVIB for a pleasantness rating, then as VIS_{EMPH} for Urgency. Alternatively, we might develop an improved proxy vibration by combining modalities - a visualization with an accompanying low-fidelity vibration.

6.8.3 Animate visualizations (G3:promising directions)

Speed and Urgency were not as effectively transmitted with our visualizations as with our vibration. Nor was Duration well portrayed with VIS_{DIR}, which had a shorter time axis than the exaggerated VIS_{EMPH}. It may be more difficult for visual representations to portray time effectively: perhaps it is hard for users to distinguish Speed/Urgency, or the time axis is not at an effective granularity. Animations (e.g., adding a moving line to help indicate speed and urgency), might help to decouple these features. As with triangulation, this might also be accomplished through multimodal proxies which augment a visualization with a time-varying sense using sounds or vibration. Note, however, that Duration was more accurately portrayed by VIS_{EMPH}, suggesting that direct representation of physical features *can* be translated.

6.8.4 Sound could represent Energy (G3:promising directions)

Our high-fidelity reference is a voice-coil actuator, also used in audio applications. Indeed, in initial pilots we played vibration sound files through speakers. Sound is the closest to vibration in the literature, and a vibration signal's sound output is correlated with the vibration energy and sensation.

However, in our pilots, sometimes the vibration sound did not match the sensation; was not audible (low frequency vibrations); or the C2 could only play part of the sound (i.e, the sound was louder than the sensation).

Thus, while the raw sound files are not directly translatable, a sound proxy definitely has potential. It could, for example, supplement where the VIS_{DIR} waveform failed to perform well on any metric (aside from Duration) but a more expressive visual proxy (VIS_{EMPH}) performed better.

6.8.5 Device dependency and need for

Energy model for Vibrations (G4:challenges) Energy did not translate well. This could be a linguistic confusion, but also a failure to translate this feature. For the visualization proxies, it may be a matter of finding the right representation, which we continue to work on.

However, with LOFIVIB, this represents a more fundamental tradeoff due to characteristics of phone actuators, which have less control over energy output than we do with a dedicated and more powerful C2 tactor. The highest vibration energy available in phones is lower than for the C2; this additional power obviously extends expressive range. Furthermore, vibration energy and time are coupled in phone actuators: the less time the actuator is on, the lower the vibration energy. As a result, it is difficult to have a very short pulses with very high energy (V1,V3,V8). The C2's voice coil technology does not have this duty-cycle derived coupling. Finally, the granularity of the energy dimension is coarser for phone actuators. This results in a tradeoff for designing (for example) a ramp sensation: if you aim for accurate timing, the resulting vibration would have a lower energy (V10). If you match the energy, the vibration will be longer.

Knowing these tradeoffs, designers and researchers can adjust their designs to obtain more accurate results on their intended metric. Perhaps multiple LOFIVIB translations can be developed which maintain different qualities (one optimized on timing and rhythm, the other on energy). In both these cases, accurate models for rendering these features will be essential.

6.8.6 VT affective ratings are generally noisy (G4:challenges)

Taken as a group, participants were not highly consistent among one another when rating these affective studies, whether local or remote. This is in line with previous work [201], and highlights a need to further develop rating scales for affective touch. Larger sample sizes, perhaps gathered through crowdsourcing, may help reduce or characterize this error. Alternatively, it gives support to the need to develop mechanisms for individual customization. If there are "types" of users who do share preferences and interpretations, crowdsourcing can help with this as well.

6.8.7 Response & data quality for MTurk LOFIVIB vibrations (G4:challenges)

When deploying vibrations over MTurk, 8/29 participants (approximately 31%) completed the survey using non-Android based OSes (Mac OS X, Windows 7,8.1, NT) despite these requirements being listed in the HIT and the survey. One participant reported not being able to feel the vibrations despite using an Android phone. This suggests that enforcing a remote survey to be taken on the phone is challenging, and that additional screens are needed to identify participants not on a particular platform. Future work might investigate additional diagnostic tools to ensure that vibrations are being generated, through programmatic screening of platforms, well-worded questions and instructions, and (possibly) ways of detecting vibrations actually being played, perhaps through the microphone or accelerometer).

6.8.8 Automatic translation (G4:challenges)

Our proxy vibrations were developed by hand, to focus on the feasibility of crowdsourcing. However, this additional effort poses a barrier for designers that might negate the benefits of using a platform of MTurk. As this approach becomes better defined, we anticipate automatic translation heuristics for proxy vibrations using validated algorithms. Although these might be challenging to develop for emotional features, physical properties like amplitude, frequency, or measures of energy and roughness would be a suitable first step. Indeed, crowdsourcing itself could be used to create these algorithms, as several candidates could be developed, their proxy vibrations deployed on MTurk, and the most promising algorithms later validated in lab.

6.8.9 Limitations

A potential confound was introduced by VIS_{EMPH} having a longer time axis than VIS_{DIR}: some of VIS_{EMPH}'s improvements could be due to seeing temporal features in higher resolution. This is exacerbated by V10 being notably longer than the next longest vibration, V8 (13.5s vs. 4.4s), further reducing temporal resolution vibrations other than V10.

We presented ratings to participants by-vibration rather than by-rating. Because participants generated all ratings for a single vibration at the same time, it is possible there are correlations between the different metrics. We chose this arrangement because piloting suggested it was less cognitively demanding than presenting metrics separately for each vibration. Future work can help decide whether correlations exist between metrics, and whether these are an artifact of stimulus presentation or an underlying aspect of the touch aesthetic.

Despite MTurk's ability to recruit more participants, we used the same sample size of 40 across both studies. While our proxies seemed viable for remote deployment, there were many unknown factors in MTurk user behaviour at the time of deployment. We could not justify more effort without experiencing these factors firsthand. Thus, we decided to use a minimal sample size for the MTurk study that was statistically comparable to the local studies. In order to justify a larger remote sample size in the future, we believe it is best to iterate the rating scales and to test different sets of candidate modalities.

As discussed, we investigated two proxy modalities in this first examination but look forward to examining others (sound, text, or video) alone or in combination.

6.9 Conclusion

In this paper, we crowdsourced high-level parameter feedback on VT sensations using a new method of *proxy vibrations*. We translated our initial set of high-fidelity vibrations, suitable for wearables or other haptic interactions, into two proxy modalities: a new VT visualization method, and low-fidelity vibrations on phones.

We established the most high-risk aspects of VT proxies, namely feasibility in conveying affective properties, and consistent local and remote deployment with two user studies. Finally, we highlighted promising directions and challenges of VT proxies, to guide future tactile crowdsourcing developments, targeted to empower VT designers with the benefits crowdsourcing brings.

6.10 Acknowledgments

We are grateful for participant feedback, reviewer suggestions, and extra tactors shared by Hong Tan. Research was funded by NSERC and conducted under UBC BREB H13-01646.

Chapter 7

Breadth: Focused Design Projects

In Chapter 7, we complement the vibrotactile tools and techniques in Chapters 3-6, broadening our scope to include application areas like gaming and education, non-vibrotactile haptic devices, and concepts like customization and other methods of *sharing*. We adopt a haptic designer's role to gain first-hand knowledge into HaXD in a more natural design setting than our one-session lab-based evaluations. We include five design projects:

- **7.1 FeelCraft: Sharing Customized Effects for Games**[1,2], a plug-in architecture for distributing customizable feel effects, implemented with the game Minecraft.
- **7.2 Feel Messenger: Expressive Effects with Commodity Systems**[3,4], a design project creating expressive shareable VT icons on commodity smart phones.
- **7.3 RoughSketch: Designing for an Alternative Modality**, a drawing application using programmable friction with the TPad phone.
- **7.4 HandsOn: Designing Force-Feedback for Education**[5], a conceptual model for DIY force-feedback haptics in education.

[1] Schneider, Zhao, and Israr. (2015) *FeelCraft: User-Crafted Tactile Content*. Lecture Notes in Electrical Engineering 277: Haptic Interaction.

[2] Zhao, Schneider, Klatzky, Lehman, and Israr. (2014) *FeelCraft: Crafting Tactile Experiences for Media using a Feel Effect Library*. UIST '14 Demos.

[3] Israr, Zhao, and Schneider. (2015) *Exploring Embedded Haptics for Social Networking and Interactions*. Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '15.

[4] Schneider, Zhao, and Israr. (2015) *Feel Messenger: Embedded Haptics for Social Networking*. World Haptics '15 Demos.

[5] Minaker, Schneider, Davis, and MacLean. (2016) *HandsOn: Enabling Embodied, Creative STEM e-learning with Programming-Free Force Feedback*. Haptics: Perception, Devices, Control,

7.5 **CuddleBit Design Tools: Sketching and Refining Affective Robot Behaviours**[6] Voodle and MacaronBit are design tools for CuddleBits, simple affective robots.

Sections 7.1-7.3 and 7.5 were primarily presented as demos with associated papers; we present these in a summary format rather than full reproduction. Section 7.4 is fully reproduced with a preface.

7.1 FeelCraft: Sharing Customized Effects for Games

As shown in prior work [200**?**], and as we will discuss in Chapter 8, customization is an important feature for haptic experiences. In addition, haptic media must be built around existing infrastructure, as it is not directly supported by most media types. FeelCraft is a media plugin architecture that monitors events and activities in the media, and associates them to user-defined haptic content in a seamless, structured way. The FeelCraft plugin allows novice users to generate, recall, save, and share haptic content, and play and broadcast them to other users to feel the same haptic experience, without having any skill in haptic content generation. In this chapter, we describe the plug-in architecture, envisioned applications, and our implementation for VT grid arrays displaying Feel Effects (FEs) [108] for a popular video game, Minecraft. Our implementations uses the Marvel Avengers Vybe Haptic Gaming Pad by Comfort Research (http://comfortresearch.com), a chair-shaped pad with 12 actuators (6 voice coils and 6 rumble motors). We designed effects that leveraged this display, *e.g.*, voice coils simulating rain on the user's back when there is rain in-game, and rumble motors creating a galloping sensation on the chair's seat when the user rides an virtual horse.

7.1.1 FeelCraft Plugin and Architecture

A FeelCraft plugin maps media to haptic sensations in a modular fashion, supporting arbitrary media types and output devices. By using a FeelCraft plugin, users can link existing and new media to the haptic feedback technology, use an FE library to find appropriate semantically defined effects, author, customize, and share a common, evolving repository of FEs, and play and broadcast haptic experiences to one or more user(s). A pictorial description of the FeelCraft architecture is shown in Fig. 1 architecture.

---

and Applications: 10th International Conference, EuroHaptics 2016, London, UK, July 4-7, 2016, Proceedings, Part II.

[6]Bucci, Cang, Chun, Marino, Schneider, Seifi, and MacLean. (2016) *CuddleBits: an iterative prototyping platform for complex haptic display*. EuroHaptics '16 Demos.

The conceptual framework of FeelCraft revolves around the FE library introduced in [108]. The FE library provides a structured and semantically correct association of media events with haptic feedback. By using the authoring interface to tailor FE parameters, a repository of FEs can remain general while being used for unique, engaging, and suitable sensations for different media. The playback system, authoring and control interface, Event2Haptic mappings, and media plugin support seamless flow of the media content to the haptic feedback hardware.

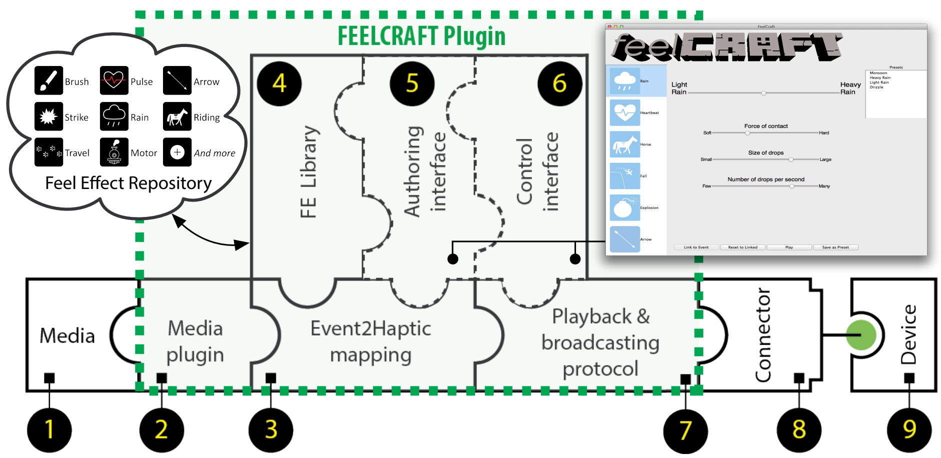


Figure 7.1: FeelCraft architecture. The FeelCraft plugin is highlighted in green. The FE library can connect to shared feel effect repositories to download or upload new FEs. A screenshot of our combined authoring and control interface is on the right.

Media (1) can be entertaining, such as video games, movies, and music, or social and educational. The media can also be general user activity or embedded events in applications. In our implementation (Figure 7.3), we use the popular sandbox indie game Minecraft (https://minecraft.net).

Media Plugin (2) is a software plugin that communicates with the media and outputs events and activities. This plugin can be as simple as receiving messages from the media or as complicated as extracting events and activities from a sound stream. With existing media, common plugin systems are automatic capture of semantic content from video frames [160], camera angles [57], or sounds [36, 131], or the interception of input devices (such as game controllers or keyboard events). We use a CraftBukkit Minecraft server modification to capture in-game events.

Event2Haptic (3) mappings associate events to FEs, which are designed, tuned, and approved by users using the FE library. This critical component links the media plugin's output to the haptic playback system. Currently, six FEs are triggered by six recurring in-game events: the presence of rain, low player health, movement on horse, strike from a projectile, in-game explosions, and player falls. Our implementation provides the option to store this mapping directly in the source code, or in a text-based JavaScript Object Notation (JSON) file

FE Library (4) is a collection of FEs. A key feature of an FE is that it correlates the semantic interpretation of an event with the parametric composition of the sensation in terms of physical variables, such as intensity, duration, and temporal onsets [108]. Each FE is associated with a family, and semantically, similar FEs are associated with the same family. For example, the Rain family contains FEs of light rain and heavy rain; as well that that of sprinkle, drizzle, downpour, and rain.In our implementation, each FE family is represented as a Python source file that defines parametric composition of the FE and playback sequences for the FeelCraft Playback system, and each FE is coded as preset parameters in a JSON file. FE family files are necessary to play corresponding FEs in the family, and new FE families can be developed or downloaded through the shared FE repository. The FE can also be created, stored, and shared. FE family and FE files are stored in a local directory of the plugin and loaded into FeelCraft on startup.

Authoring and Control Interfaces (5, 6) allow users to create and save new FEs and tune, edit, and play back existing FEs. Users modify an FE by varying sliders labeled as common language phrases instead of parameters such as duration and intensity (Fig. 1). Therefore, users can design and alter FEs by only using the semantic logic defining the event. The interface also allows users to map game events to new FEs and broadcast to other users, supporting a What-You-Feel- Is-What-I-Feel (WYFIWIF) interface [191].

Playback and Communication Protocols (7) render FEs using the structure defined in FE family files and outputs them through a communication method (8) to one or more devices (9). Our implementation includes an API controlling the commercially available Vybe Haptic Gaming Pad via USB.

7.1.2 Application Ecosystem

FeelCraft plugins are designed to make haptics accessible to end users using existing media and technology. For example, a user may want to assign a custom

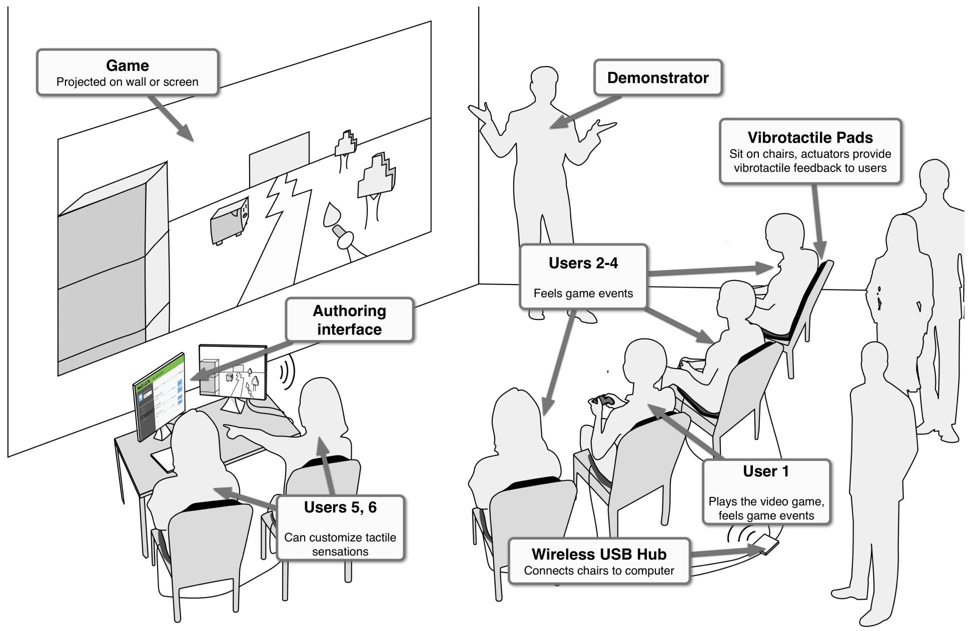


Figure 7.2: Mockup for FeelCraft demo system.

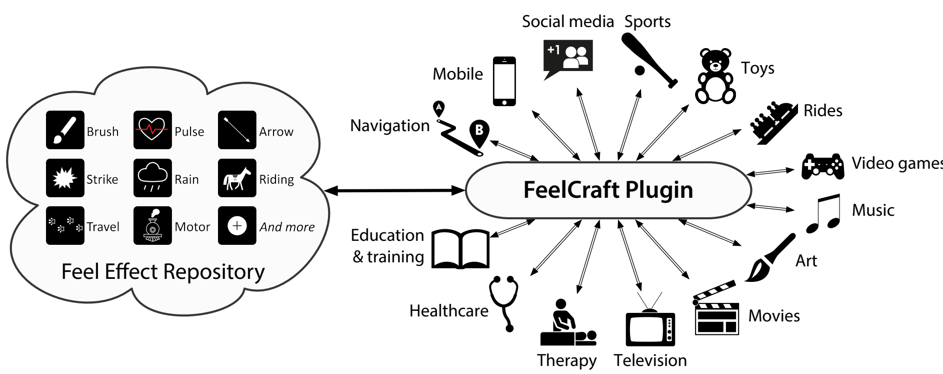


Figure 7.3: Application ecosystem for FeelCraft and an FE repository

vibration to a friend's phone number, or add haptics to a game. In this case, a user would download a FeelCraft plugin for their device, browse FEs on an online feel repository, and download FE families they prefer. Once downloaded, the FeelCraft authoring interface allows for customization, as a rain FE for one video game may not quite suit another game. The user could create a new FE for their specific application, and once they were happy with it, upload their custom FE for others to use. If the user wanted to show a friend their FE, they could use the playback system to drive output to multiple devices, or export the FE to a file and send it to them later. Figure 2 illustrates this ecosystem with application areas. Just like the Noun Project for visual icons (http:// thenounproject.com) and downloadable sound effect libraries, we envision online repositories of FEs that can be continually expanded with new FEs by users. Our current FErepository includes six original families described in [108] and an additional four new families: Ride, Explosion, Fall, and Arrow.

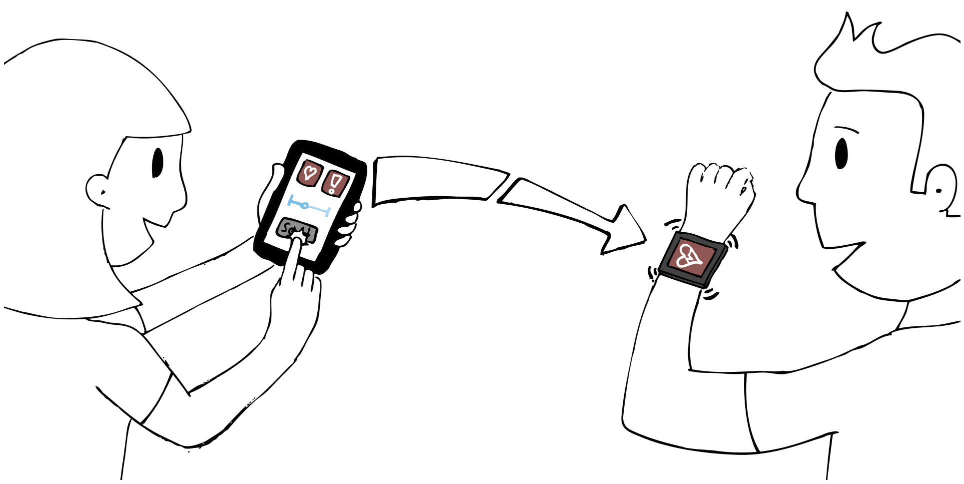


Figure 7.4: Users exchanging expressive haptic messages on consumer embedded devices.

7.2 Feel Messenger: Expressive Effects with Commodity Systems

In Section 7.1, we designed expressive spatial VT Feel Effects using existing infrastructure, using a plugin architecture to link desktop applications to new VT hardware. In Section 7.2, we look at the expressiveness of existing infrastructure and actuation methods with Android smartphones for customizable VT effects by implementing customizable VT emojis in a chat program, Feel Messenger. As explained previously by Figure 6.6 in Chapter 6, APIs for vibrotactile feedback are limited to a series of pulses. With Feel Messenger, we were able to produce expressive VT icons for emojis using the built-in Android API, including customizable effects like a heartbeat that varies in both rate and intensity. We found VT icons were effective when they had an engaging visual icon to frame the vibration, *e.g.*, a cat emoji helped the user understand purring vibration was a purr. Figure 7.4 presents a concept sketch.

7.2.1 Feel Messenger Application

In this section, we present the architecture (backend) and user interface (frontend) of a messenger application that allows users to create and share haptic content through a network connection. The critical components of the application are shown in Figure 2.

Architecture – To account for the limited computation, storage and communication capabilities of a simple microcontroller unit, we introduce feelbits and feelgits. Feelgits (short from "feel widgets") are installed piece of software that define parametric compositions of a set of haptic patterns (called a family). Feelbits are parametric settings of a feelgit to produce a particular haptic pattern (called a feel effect [108]). For example, the feelgit of pulse is defined as two successive onsets of vibration, separated by a timing parameter. The feelbits are timing and intensity of onset parameters. Therefore, by varying feelbits, a user can personalize the haptic effect to be calm (low intensity, long temporal separation) or racing (high intensity, short temporal separation) heartbeat.

A library of haptic patterns is stored as parametric models (feelgits) with preset parameters (feelbits). New feelgits and feelbits can be downloaded, personalized and saved. The haptic engine idly waits for incoming haptic messages and renders haptic patterns on demand. Once the message is received, the corresponding feelgit is executed with parameters defined as feelbits. Once the pattern is completely rendered, the engine waits idly for the next message.

Additionally, the response characteristics of the VT motor are also stored in the memory. These characteristics are generally represented by simple first-order functions relating the digital value (such as data byte) to the perceived intensity judged by users [116], which could be used to maintain the quality of experience across wide variety of mobile phones and hardware technologies.

Finally, we introduce a communication protocol that shares feelgits and feelbits along with text messages. For example, the frontend application sends a function playpattern("pulse", p1, p2) to play the feelgit pulse with parameters defined as feelbits p1 and p2; or playpulse("soft") plays a predefine soft pulse. Note that in order for the device to play a haptic pattern, the corresponding feelgit must be stored in the device; the communication packet includes feelbits and the name or id of the corresponding feelgit.

Predefined Patterns – The predefined patterns allow users to quickly attach a haptic pattern to the IM. These patterns can be stored from incoming messages or created by using stored haptic families. Each pattern is defined by a set of feelbits that plays when the corresponding feelgit is executed. These presets can be shown as text, images or emotion icons.

Authoring interface – The Feel Editor displays available FE families (feelgits) and allows users to personalize, play, save and share haptic patterns. By clicking a FE icon, sliders corresponding to parameters (feelbits) are enabled. These sliders may have labels corresponding to physical parameters, such as amplitude or duration of vibration; however, we have used semantic labeling that may correspond to single or multiple parameters. Once the sliders are adjusted, the user can play, save or attach the haptic pattern to the IM.

7.2.2 Haptic Vocabulary

The vocabulary of haptic effect is critical for expressive and precise communication between users. In this preliminary implementation, we explore three types of haptic vocabularies. Type 1 is adapted from feel effects defined in [108], where haptic patterns are semantically characterize by a phrase. Type 2 is change in physical parameters as in [17, 140] but can also be simultaneously played with feel effects. Type 3 predefined coded patterns. Figure 7.5 shows the icons for haptic language. Note, that the two feel effects cannot be simultaneously played. This will result in overflow of the user's bandwidth, especially with a low-fidelity VT actuator.

Type 1: Feel Effects – A set of feel effects is defined that delivers emotional, attentional and contextual effects. They are:

Pulse: Two successive onsets of vibration; speed (slow/fast) and intensity (weak/strong). Used as pulsation and heartbeat (calm/racing).

Motor: A 4-second modulated vibration; intensity (soft/loud) and speed (slow/-fast) are parameters. Used as snoring, breathing, purring, engine rumble, etc.

Strike: A single onset of vibration; duration (short/long) and intensity of vibration are parameters. Used for tap, poke, jab and punch.

Urgency: a burst of vibrations; intensity (weak/strong) and temporal separation between pulses (low/high urgency) are parameters. Used for alerting users and expressing urgency.

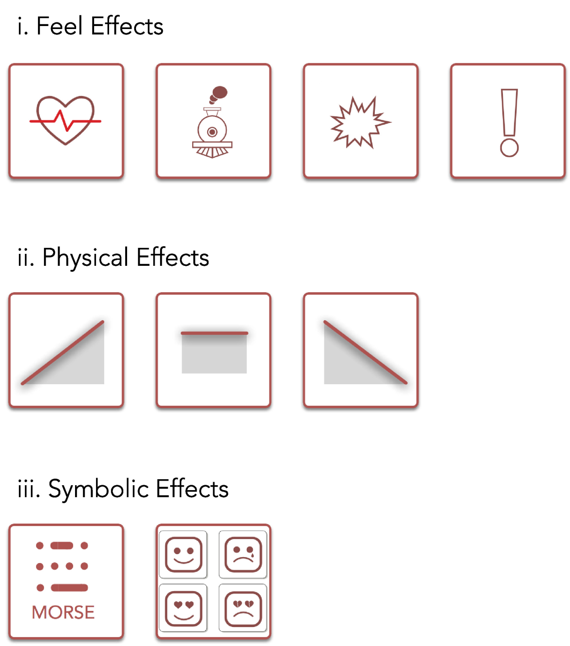


Figure 7.5: Graphical representation of haptic vocabularies and icons.

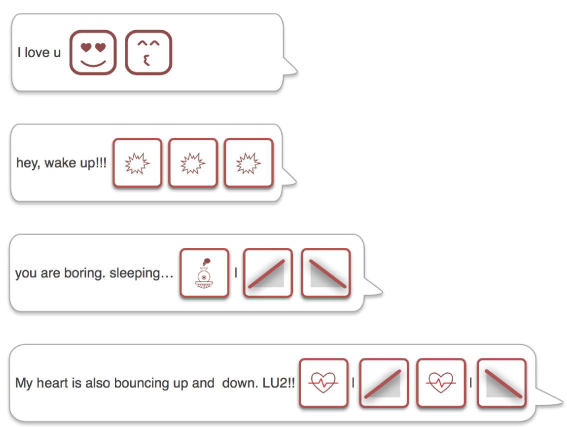


Figure 7.6: Some examples of expressive haptic messages embedded with normal text messages.

Type 2: Physical Effects – These effects are associated with direct variation in tactile patterns. Previous studies (*e.g.*, [17, 140]) have used variation in amplitude, duration as typical variation. Our library includes:

Ramp-up: gradual increase in intensity; parameters are peak intensity and the rate of increase.

Ramp-down: gradual decrease in intensity; parameters are peak intensity and the rate of decrease.

Spacer: keeps steady intensity; parameters are intensity and duration. This can be used for putting a delay (or spaces) between two haptic effects.

These effects create new haptic effects and can also be combined with feel effects. Such as the message Ramp-up — Motor followed by Ramp-down creates a new pattern that gradually increases the rumbling and then decays linearly as shown in Figure 7.6.

Type 3: Coded Effects – This type demonstrates symbolic vocabularies, such as one adapted from International Morse Code that consists of pre-stored pulses of dots and dashes. Other examples can be vibratese language [219], emoticon, and input from peripheral sensors.

7.2.3 Demo

We developed an Android application on a two Samsung S5 smartphones running Android 4.4.2. The Android API allows ON/OFF control of the embedded VT motor. A rough relationship between duty cycle and perceived intensity was determined to create effects.

In this prototype, we explored both predefined effects and Type 1 icons (Feel Effects). Our predefined effects were designed with 6 emoji. Our four Type 1 FEs were: Heartbeat (Pulse FE), Lightning (Strike FE), Cat Purr (Motor FE), and Coffee (Urgency FE). These VT emoji could be embedded in chat messages, sent between two Android phones using UDP. VT effects are felt when editing, when received, and when the user taps a message. All effects were implemented using built-in Android APIs.

7.3 RoughSketch: Designing for an Alternative Modality

FeelCraft (Section 7.1) and Feel Messenger (Section 7.2) used existing infrastructure to conduct VT design, but how do other types of feedback vary? In Sections 7.3 to 7.5, we investigate other modalities in other applications: programmable

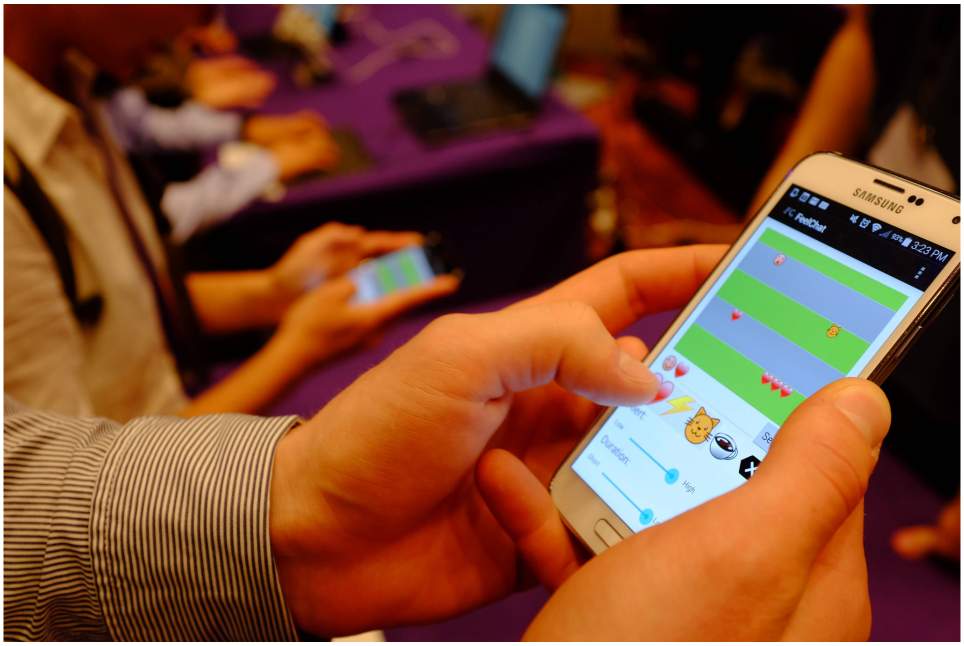


Figure 7.7: Implemented Feel Messenger demo at World Haptics 2015.

friction for touchscreen drawing, force-feedback for education, and affective robots for emotional expression. Here, in Section 7.3, we describe RoughSketch, a drawing application for the TPad Phone.

The TPad Phone (www.thetpadphone.com) is a programmable friction display mounted on an Android phone. It uses piezo-actuated mechanical vibration to create a cushion of air, reducing friction [232]. As part of the World Haptics 2015 Student Innovation Challenge, we built RoughSketch, a mobile drawing application to explore friction displays for digital mark-making. We looked at several mark-making interaction techniques including:

- *Paintbrush*, where you feel paint leaving your finger,
- *Pen and eraser*, based on real-world writing utensils,
- *Spray paint*, where you feel the roughness of paint on the screen as you spray,
- *Pinch/zoom*, inspired by compressing and stretching rubber, and
- *Feel finger*, the ability to feel your drawing on the paper.

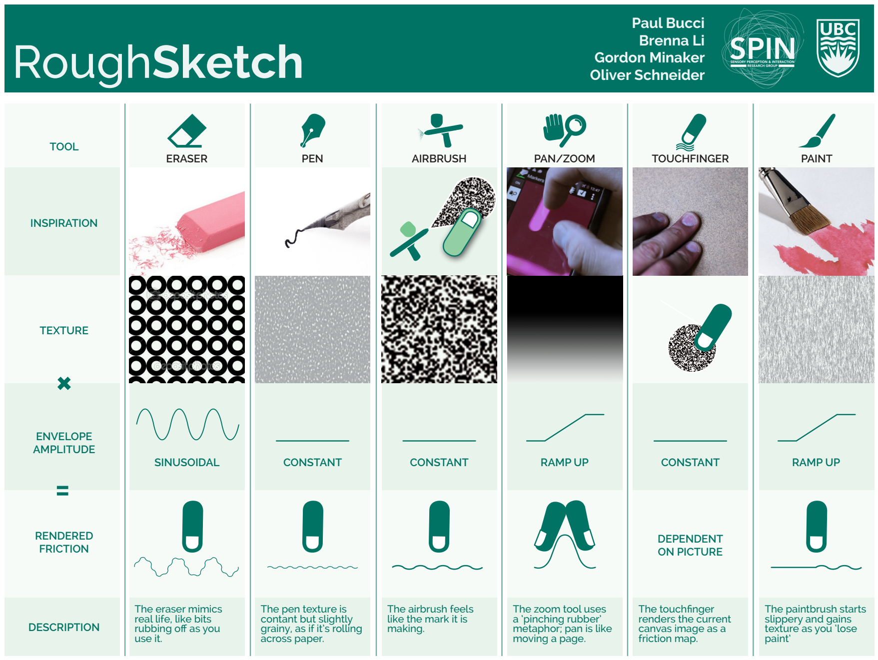


Figure 7.8: RoughSketch handout, illustrating interaction techniques an textures.

To implement RoughSketch, we adapted an open-source Android drawing application, Markers (https://github.com/dsandler/markers) and used the TPad Phone API to control friction using two methods: static textures defined by bitmaps, and temporal envelopes that programmatically adapt friction based on input values or time. We used a variety of real-world metaphors to inspire our designs; these are illustrated in Figure 7.8. While designing and developing RoughSketch, we exposed a design space, finding conflicts for our metaphors, specifically, should TPad sensations feel like their real-world equivalent, or are they unique to the TPad; and should rendered textures represent the drawing process, or the finished product?

Our findings are outlined in Figure 7.9. In addition to developing different effects, we informally compared haptic feedback to non haptic feedback by including a toggle to friction feedback. Although some effects were subtle, once disabled, users immediately noticed the difference and preferred to have haptic feedback. We also explored stylus use, finding that a rubber tip would barely transmit any sensation, while a more rigid tip would propagate the (dampened) effect.

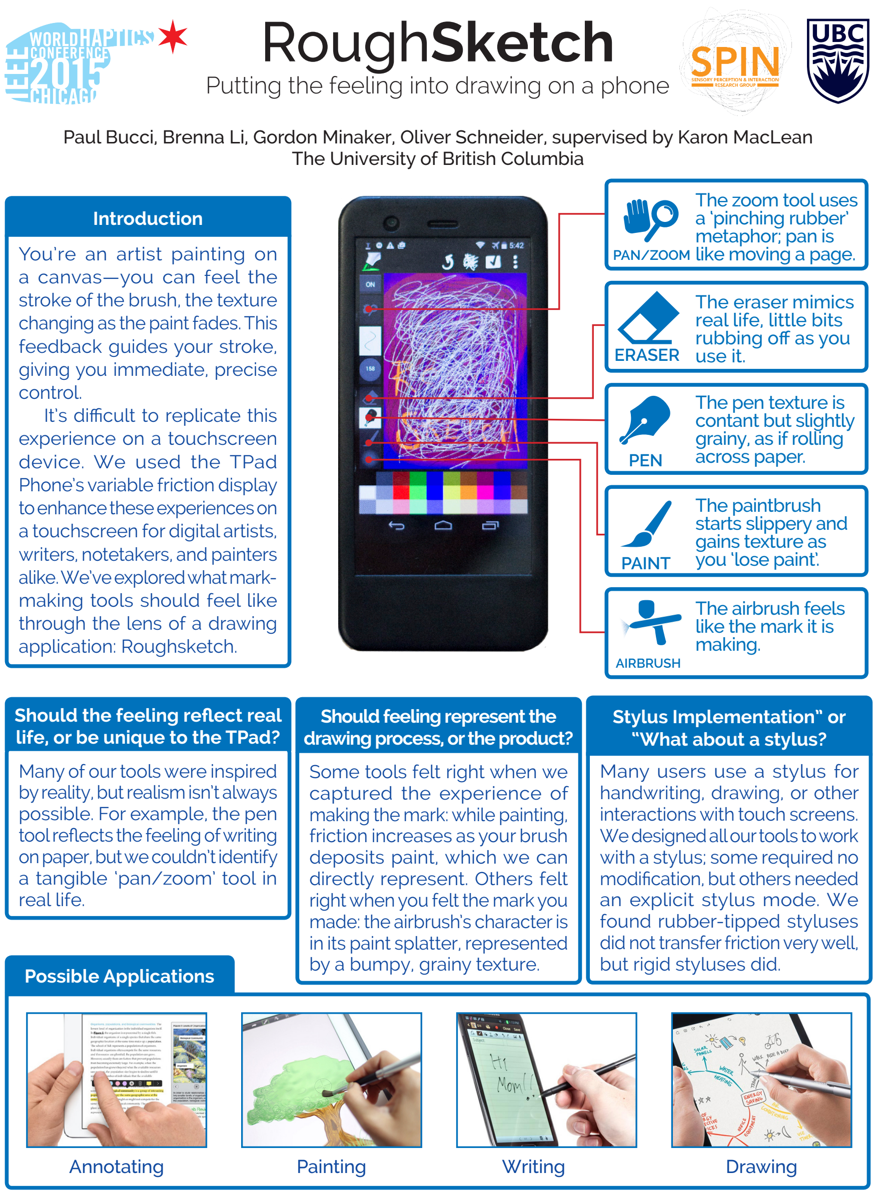


Figure 7.9: RoughSketch poster, describing interaction techniques and high-level findings.

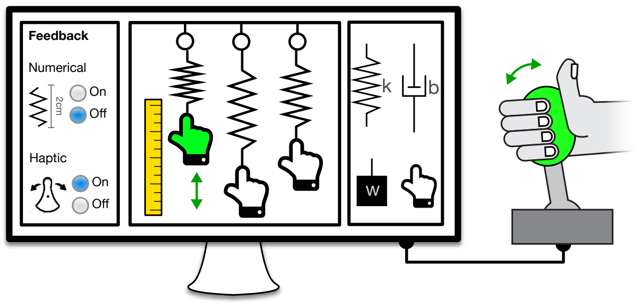


Figure 7.10: Students, teachers, and researchers can explore science, technology, engineering, and math (STEM) abstractions through low-fidelity haptics, incorporating elements into system designs.

7.4 HandsOn: Designing Force-Feedback for Education

Preface – In Section 7.4, we investigate creative control of 1-degree of freedom (DOF) force-feedback display for education. Force-feedback is interactive, with output dependent on input, but also controllable when the user holds their hand stationary, unlike programmable friction feedback explored in Section 7.3. The application area, science education, offers important design constraints: feedback must enhance learning without distraction, and in this project, enable creative exploration for students. We thus both design haptic feedback and enable students to design while they learn. To manage this, we model feedback as a system of springs, easy to adjust and design, but scalable to more complex tasks by combining multiple springs in series or parallel.

7.4.1 Abstract

Embodied, physical interaction can improve learning by making abstractions concrete, while online courses and interactive lesson plans have increased education access and versatility. Haptic technology could integrate these benefits, but requires both low-cost hardware (recently enabled by low-cost DIY devices) and accessible software that enables students to creatively explore haptic environments without writing code. To investigate haptic e-learning without user programming, we developed *HandsOn*, a conceptual model for exploratory, embodied STEM education software; and implemented it with the *SpringSim* interface and a task battery for high school students. In two studies, we confirm that low-cost devices can render haptics adequately for this purpose, find qualitative impact of *SpringSim* on student strategies and curiosity, and identify directions for tool improvement and extension.

7.4.2 Introduction

Recognition of the value of a hands-on, embodied approach to learning dates to 1907, when Maria Montessori opened a school where she used *manipulatives* to teach a wide array of concepts ranging from mathematics to reading, e.g., by introducing the alphabet through children tracing their finger along large, cut-out letters [154]. Constructivist learning theories posit that well-designed manipulatives can assist understanding by grounding abstract concepts in concrete representations [**? ?**], and are an accepted core principle in early math and science education, confirmed empirically [31].

More recently, digital technologies are radically altering learning environments. Massive Open Online Courses (MOOCs) expand access, games motivate, and with graphical simulations (e.g., PhET [229]), students can interact with abstractions to develop their understanding. However, these experiences are disembodied. Indirect contact via keyboard, mouse and screen introduces a barrier of abstraction that undermines the connection and path to understanding.

Haptic (touch-based) technology should bring benefits of physicality and embodied learning [59] to interactive virtual environments. It adds a sensory channel as another route to understanding [28]; when deployed appropriately, active exploration can improve understanding [147] and memory [82] of new concepts. Haptic tools have already shown promising results in many specializations, demographics and age groups, both to enhance lesson fidelity and to increase engagement and motivation through tangibility and interactivity; e.g., with devices like Geomagic Touch[7] [230] and SPIDAR-G [186].

Unfortunately, existing approaches have both hardware and software limitations. Actuated learning tools introduce physical issues of cost, storage, and breakage; devices are too bulky, complex, or expensive for schools or self-learners. For software, it is hard for users to construct and explore their own haptic environments. Typically, users load a virtual system to interact with it haptically. This sidelines the rich learning potential of involving users with model construction [**?**]. We address hardware with the HapKit [168], a $50, simple, low-fidelity device constructed from 3d printed materials.

Our focus here is on software, with a new learning environment that lets users both construct and explore haptic systems. Until now, the only way for a user to construct a haptic system was by programming it herself. Our approach, inspired by Logo [**?**] and Scratch [145], is to ultimately provide much of the power of a programming language while hiding distracting complexity.

[7] Prev. Sensable Phantom www.geomagic.com/en/products/phantom-omni/overview

Approach and Present Objectives:

To study *how* to unlock the potential of hapticized virtual environments in STEM education, we need a viable front-end. To this end, we first established a *conceptual model* (*HandsOn*): central interface concepts, supported operations and language [113] that can be employed in a broad range of lessons involving physical exploration and design.

Next, we implemented the *HandsOn* conceptual model (CM) in *SpringSim*, a first-generation learning interface prototype narrowly focused in a module on mechanical springs and targeted at high school physics students. To render forces we used the HapKit, a simple device with a 3D-printable handle providing affordable, self-assembled 1 DOF force-feedback for about $50 USD. As an evaluation instrument, this single-lesson implementation allows us to (a) measure a given hardware platform's fidelity for a representative perceptual task; (b) attain insight into the kinds of lessons such a system can leverage; and (c) assess its learning-outcome efficacy relative to conventional methods. With these answers, we will be able to design a more powerful tool.

We report results from two user studies: (1) the HapKit's ability to display differentiable springs with and without graphical reinforcement, and (2) a qualitative evaluation of *SpringSim* for a carefully designed set of educational tasks. We confirm that the *SpringSim* interface and its conceptual model *HandsOn* are understandable and usable, describe the role of haptics compared to mouse input, and provide recommendations for future evaluation, lesson and tool design.

7.4.3 Tool Development: Conceptual Model and Interface

Our goal was to find a software model to use and evaluate low-cost force feedback in an educational setting. We began by choosing a device, establishing requirements, and exploring capabilities through use cases and prototypes. From this, we defined *HandsOn*. We then implemented essential features in a medium-fidelity prototype, *SpringSim*, for our user studies.

Initial design (requirements):

We established six guiding requirements. First, we developed initial prototypes with HapKit 2.0 through two pilot studies with middle school students (described in [168]). These highlighted two aspects of a practical, accessible approach for junior students: 1) no programming; instead 2) a graphical implementation of an exploratory interface within a lesson plan. We also needed to build on known benefits of traditional classroom practices, and enable learning-outcome comparison. We must 3) support the same *types* of traditional education tasks, e.g., let students compare and assemble spring networks as easily as in a hands-on physics lab; but also 4) *extend* them, to leverage the flexibility offered by a manipulative that is also virtual. Similarly, to support future formal comparisons, our model needs to 5) support both haptic and non-haptic (mouse) inputs. Finally, to ensure generality we also needed to 6) support diverse STEM topics, like physics, biology, and mathematics. Further design yielded a model that addressed these requirements: *HandsOn*.

Conceptual Model:

HandsOn is a programming-free (R1) graphical interface supporting learner exploration (R2), with a number of key *concepts*: *Interactive Playground*, *Hands*, *Design Palette*, *Objects*, *Properties*, *Haptic* and *Visual Controls*. Exploration is supported at various levels (Figure 7.11).

The *Interactive Playground* provides a virtual sandbox where users can interact with virtual environments (VE). *Hands* allow users to select, move, and manipulate components in the Interactive Playground. Control occurs with either the mouse or a haptic device to receive force-feedback (Figure 7.11A) (R5). In the design and modification phase, users can add or remove *objects* like springs, masses, gears, or electrons by dragging them to and from a *Design Palette* (R3). Once added to the scene, users can modify their physical properties (e.g., a spring constant k) and make changes to the VE (Figure 7.11B). After construction, the user can customize their interaction with their VE by adjusting *Visual Controls* and *Haptic Controls* options that extend interactions in new ways afforded by haptics (R4) (Figure 7.11C). Because of the flexibility afforded by having multiple *objects* in the playground with multiple *Hands* for interaction points, and customization of interaction and feedback, *HandsOn* can support different STEM topics (R6), from biology to mathematics. To confirm the viability of this approach, we built an

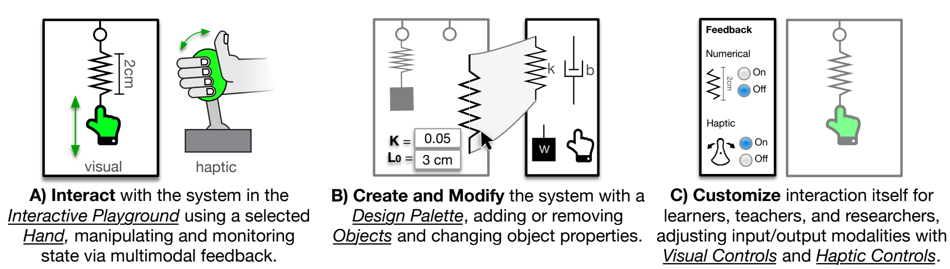


Figure 7.11: The *HandsOn* CM enables three kinds of exploration based on requirements.

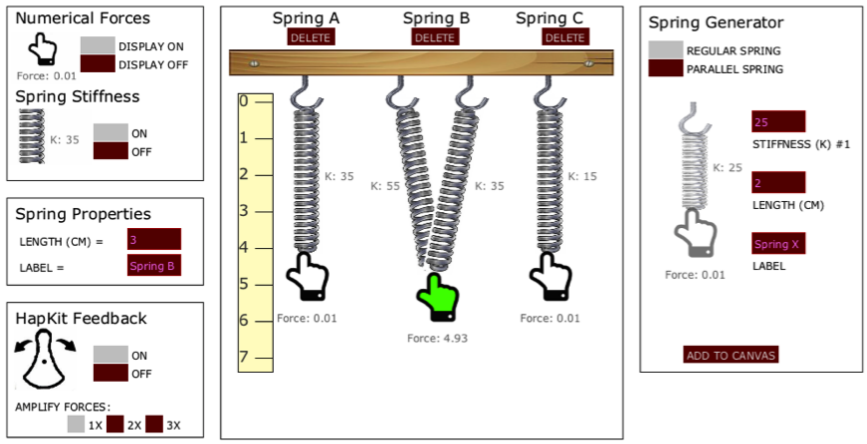


Figure 7.12: *SpringSim* interface, a *HandsOn* sandbox for a single lesson module on springs.

initial prototype with essential features: *SpringSim*.

Implemented Prototype:

Our first *HandsOn* interface is *SpringSim* (Figure 7.12), which supports a spring lesson – spring systems are natural as a virtual environment of easily-controlled complexity. In *SpringSim*, *objects* include single springs and parallel spring systems, with properties spring rest length (cm), stiffness (N/m) and label. The *Design Palette* includes the *Spring Properties* and *Spring Generator* UI components. Implemented *Visual Controls* are toggling numerical displays of spring stiffness and force; *Haptic Controls* toggle HapKit feedback and output amplification. The open-source repository for SpringSim is available at https://github.com/gminaker/SpringSim.

7.4.4 Study 1: Perceptual Transparency

Before evaluating *SpringSim*, we needed to confirm that the HapKit could render spring values sufficiently for our qualitative analysis.

Methods:

14 non-STEM undergraduate students (8 females) participated in a two-alternative, forced choice test with two counterbalanced within-subject conditions: *HapKit + Dynamic Graphics*, and *HapKit + Static Graphics* (Figure 7.13). Three spring pairs (15/35, 35/55 and 55/75 N/m) were each presented five times per condition,

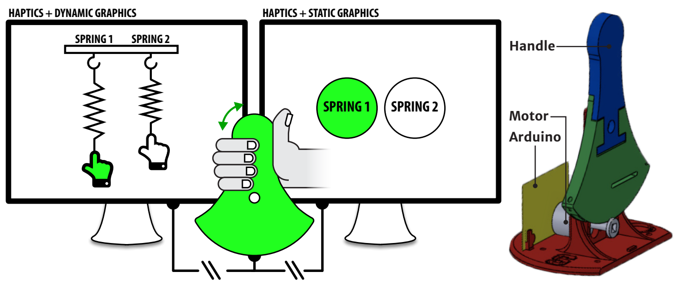


Figure 7.13: In the *Hapkit+Dynamic Graphics* condition, graphical springs responded to input (left); static images were rendered in the *Hapkit+Static Graphics* condition (right); in both, HapKit 3.0[168] was used as an input/force-feedback device (far right).

in random order. For each pair, participants indicated which spring felt more stiff, and rated task difficulty on a 20-point scale. Following each condition, participants rated overall condition difficulty, mental demand, effort, and frustration on 20-point scales derived from the NASA TLX [91]. Following the completion of both conditions, a semi-structured interview was conducted to address any critical incidents. Each session lasted 20-30 minutes.

Results:

All tests used a 5% level of significance and passed test assumptions.

Accuracy: A logistical regression model was trained on task accuracy with spring-pair and condition as factors. No interaction was detected; spring-pair was the only significant factor. Post-hoc analysis revealed that spring-pair #1 (15/35 N/m) was significantly less accurate than spring-pair #2 (35/55; p=0.0467). Performance averaged 88.57% (15/35), 96.49% (35/55), and 94.45% (55/75).

Time: Task time ranged from 3-160s (median 117s, mean 96.41s, sd 47.57s). In a 3-way ANOVA (participant, spring-pair, and visualization condition) only participant was significant ($F(13,336) = 4.17$ $p = 1.947e-06$).

Difficulty rating: A 3-way ANOVA (factors: participant, spring-pair, and visualization condition) detected one two-way interaction between participant and spring pair ($F(26,336) = 2.10$, $p = 0.00165$).

Discussion:

Study 1 revealed that (a) for stiffness intervals 15/35/55/75 N/m, the HapKit provides distinguishability equivalent to dynamic graphics. Individual differences influenced difficulty and speed, suggesting that learning interfaces may need to accommodate this variability. (b) Accuracy was not dependent on individual differences, suggesting that learning interfaces can consider task time and perceived difficulty separately from accuracy when using the HapKit (at least, for these force ranges). (c) Performance was mostly above 90%, and confidence intervals for our small sample size estimate no lower than 82% accuracy at the lowest (15/35). We speculate that the HapKit's natural dynamics are more pronounced at lower rendered forces, and may interfere with perceptibility.

7.4.5 Study 2: Tool Usability and Educational Insights

Methods:

10 non-STEM participants (1st and 2nd year university undergrads with up to first year physics training, 6 female, 17-20 years) volunteered for 45-60 minute sessions. After an introductory survey, participants were randomly assigned to one of two conditions, *Mouse* (4 participants, M1-4) or *Hapkit* (H1-6). HapKit 3.0 was calibrated for force consistency between participants. After allowing participants to freely explore *SpringSim*, a survey assessed understanding and usability of various *SpringSim* interface components; misunderstood components were clarified. Three exit surveys elicited value of *SpringSim* components on 7-point Likert scales, cognitive load [117], understanding, and curiosity on 20-point scales, and preferred learning modality [72], respectively.

Learning Tasks:

We iteratively designed and piloted a task battery of escalating learning-goal sophistication [12] to expose strategies for force feedback use and general problem-solving (Table 7.1). Tasks did not require physics knowledge, and were suitable for both mouse and HapKit input.

Analysis:

We conducted t-tests on self-reported understanding, cognitive load, engagement, understanding, curiosity; and on objective metrics of time-on-task and number of spring interactions. Qualitative analysis of video and interview data used grounded theory methods of memoing and open & closed coding [44]. Together, these

Task	Bloom	Description
1	Understand (2)	Rank three springs in order from least to most stiff
2	Understand (2)	Plot the relationship between displacement and force for two springs.
3	Apply (3)	Estimate the stiffness of an unknown spring, given two reference springs with known stiffness value
4	Analyze (4)	Predict the behaviour of springs in parallel.
5	Create (6)	Design a parallel spring system that uses two springs to behave like an individual spring of stiffness 55 N/m.
6	Apply (3)	Predict the behaviour of springs in series.
7	Evaluate (5)	Describe any relationships you have noticed between spring force, displacement, and stiffness.

Table 7.1: Learning tasks used with *SpringSim* in Study 2. *Bloom* level is a measure of learning goal sophistication [12]

yielded insight into the usability of *SpringSim* and the *HandsOn* CM, and several themes describing the role of haptics in our tasks. Two participants were excluded from analysis of Task 1 due to technical failure.

Results - Usability:

After free exploration of *SpringSim*, participants rated their understanding of CM objects (yes/no) and their ease-of-use [1-7]: *Ruler* (10/10, 7.0), *Numerical Force Display* (10/10, 6.5), *Playground* (10/10, 6.0), *Hand* (9/10, 6.0), *Spring Properties* (9/10, 6.0), *Spring Generator* (**7/10, 5.0**), *HapKit* (6/6, **4.5**), and *Haptic Feedback Controls* (5/6, **4.5**). While generally usability was good, interface clarity needed improvement in highlighted cases. Participants specifically noted confusion on radio button affordances, and *Spring Generator* input fields (due to redundant availability in *Spring properties*).

Results - Task Suitability for Haptic Research:

Regardless of prior physics knowledge, all participants were able to complete education tasks 1-6 (Table 7.1) in the allotted 60 minutes. We found no evidence that any task favoured one condition over another. When participants in the mouse condition were asked how their workflow would change with physical springs, participants weren't sure: "I don't know if that would've given me more information" (M4).

Results - Haptics & Learning Strategies:

We observed several themes relating to the influence of force feedback on a student's learning strategy.

Haptics creates new, dominating strategies. Learning strategies used by participants in the HapKit condition (H1-6) were more diverse than those in the mouse condition (M1-4). In Task 1, M1-4 all followed the same strategy, displacing all 3 springs the same distance and comparing the numerical force required to displace them. They then correctly inferred that higher forces are associated with stiffer springs (the *displace-and-compare* strategy).

By contrast, all 5 H participants included in analyses (H2 excluded due to technical failure) used force-feedback as part of their approach to Task 1. H1 describes applying the same force to the HapKit across all 3 springs, recording displacement to solve the task, while H5 described looking at the speed at which the HapKit was able to move back-and-forth in making his determination of stiffness, rather than through direct force-feedback of the device. Only H6 indicated that he "looked at the numbers for a sec", but no participant fully used the *displace-and-compare* strategy we observed for M participants.

While the single-strategy approach worked for easy tasks, it was linked to errors and dead-ends in at least one instance in the mouse condition. In Task 5, M2-4 used *displace-and-compare* to validate their newly designed spring; M1 did not seek verification of his design. In contrast, H1,2,5,6 used haptic feedback to verify their designs. They did this by comparing how stiff their parallel spring system felt to a target reference spring. H4 guessed at an answer without verification. H3 used the *displace-and-compare* strategy, checking that equal forces were required for equal displacement.

Haptic impressions of springs are enduring and transferrable. HapKit participants were able to use their previous explorations to solve problems. In Task 3, M1-4 interacted with all three springs to find a ratio between force and stiffness. However, H participants interacted with springs fewer times (mean 1.5, sd 3.21) than M (6, sd 1) ($p=0.018$). H2-4,6 did not interact with any springs, and H1 interacted with only one. This was because they had already interacted with the springs in previous questions: "I remember spring C was less stiff" (H3). Further suggesting the strength of haptic impressions, when H1 designed an inaccurate spring system for Task 5 ($k=80N/m$ vs. expected $k=55N/m$), she described the haptics as overriding the visual feedback: "they just felt similar. Even though the numbers weren't really relating to what I thought." Similarly, H2 arrived at an approximate result ($k=40N/m$), after using force-feedback and acknowledges "... [it's] slightly less than the reference spring, but it's closer."

Haptics associated with increases in self-reported curiosity and understanding. Participants' self-reported curiosity significantly increased over the course of HapKit sessions from a mean of 6.3 (sd 3.83) to 10.8 (sd 3.92) in the Hapkit condition (p=0.041). No significant changes in curiosity were detected in the mouse condition. Participants' self-reported understanding significantly increased over the course of HapKit sessions from a mean of 3.67 (sd 4.03) to 11.83 (sd 3.19) (p=0.014). No significant changes in understanding were detected in the mouse condition (before: 9.25, sd 5.32; after: 9.25, sd 5.32; p=0.77).

In interviews, participants commonly made references to how the HapKit influenced their understanding: "I can use this thing for help if I really need some physical, real-world stimuli" (H5); "almost all of my thinking was based on how the spring [HapKit] ended up reacting to it" (H6). M2, who had a stronger physics background than others (IB Physics), was the only user to report a drop in curiosity and understanding over the course of the physics tasks, despite initial excitement: "the fun part is messing around with [SpringSim]," he exclaimed near the beginning of the exploratory phase.

7.4.6 Study 2 Discussion

Tool and Tasks: Suitability for Learning and as Study Platform

Adequacy and comprehensibility of underlying model: Overall, *HandsOn* concepts proved an effective and comprehensible skeleton for *SpringSim*. Specific implementations rather than concepts themselves appeared to be the source of the reported confusions, and we observed that *HandsOn* should be extended with additional measurement tools (e.g., protractors, scales, calculators, etc).

SpringSim performance: This *SpringSim* implementation adequately supported most students in finishing learning tasks; extending available objects, properties and tasks will support advanced students as well. Future iterations should more clearly map *Design Palette* elements to the objects they support, increasing rendering fidelity and reconsider colors to avoid straightforward affordance issues. While participants did not heavily use haptic and visual controls, we anticipate these will be important for instructor and researcher use.

Learning task suitability: The learning tasks used here were fairly robust to time constraints of user-study conditions, did not require previous physics knowledge, avoided bias from standardized physics lessons, and exposed haptics utilization strategies without penalizing non-haptic controls. Currently, the task set ends by asking students to predict a serial system's behavior; some students found predicting new configurations a large jump. Future task-set iterations could support integrative, prediction-type questions with interface elements that are successively exposed to allow prediction testing.

Evidence of the Role of Force Feedback in Learning

Curiosity and understanding leading to exploration: Self-reported curiosity and understanding increased when forces were present. While these trends must be verified, curiosity is of interest since it can lead to more meaningful and self-driven interactions. Iterations on both tasks and tool should support this urge with an interface and framing that supports curiosity-driven exploration.

Alternative strategies enabled by force feedback: The HapKit's additional feedback modality enabled alternative task workflows, e.g., estimations of force appeared to supplant mathematical strategies for stiffness estimation. While possibly risky as a crutch, force assessments might be a useful step for students not ready for technical approaches (e.g., M3/Task 3 when stalled in attempting cross-multiplication). Future task-set iterations could encourage more *balanced* strategy use, e.g. mathematical *and* perceptual rather than primarily perceptual.

HapKit salience, resolution & implications: Overall, HapKit 3.0's fidelity was enough to assist participants verify a correct hypothesis. However, those who started with an *incorrect* hypothesis and used only HapKit to test it generally arrived at solutions that improved but were still inaccurate. Given the confidence that forces instilled, this is an important consideration. A formal device characterization will allow us to keep tasks within viable limits; we can also consider using low-fidelity forces more for reinforcement and exploratory scenarios.

Limitations and Next Steps:

Our studies were small and used non-STEM university students as a proxy for high-school learners. Despite both limitations, they were useful for our current needs (rich, initial feedback establishing suitability and usability for *HandsOn* through *SpringSim*); but may overestimate general academic ability and maturity. As we we move into evaluation of learning outcome impact, larger and more targeted studies are imperative.

Future interfaces can both increase physical model complexity and breadth (e.g., complex mass-spring-damper systems), and extend *HandsOn* for more abstract education topics, such as trigonometry. We also plan to extend the *Playground* to support more engaging, open-ended student design challenges, such as obstacle courses using trigonometry concepts; this in turn requires new measurement tools and tasks that are more exploratory and open-ended.

7.4.7 Conclusions

Haptic feedback's potential in STEM education use can only be accessed with a comprehensible, extendable, and transparent front-end. We present *HandsOn*, a conceptual skeleton for interfaces incorporating virtual forces into learning tasks, and assess its first implementation, *SpringSim* and task set. Our findings (on interface usability, task effectiveness, and impact of haptic feedback on learning strategies, understanding and curiosity) underscore this approach's promise, as we proceed to study haptic influence on learning outcomes themselves.

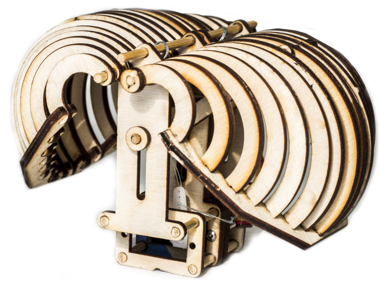
7.5 CuddleBit Design Tools: Sketching and Refining Affective Robot Behaviours

In Section 7.5, we explore a third non-vibrotactile modality - furry, affective, breathing robots called CuddleBits (Figure 7.14). These robots are multimodal: they visibly move, and their breathing can be felt. Their form factor and affordances can vary; for example, they can be flexible (Figure 7.14a) or rigid (Figure 7.14b). We use the CuddleBits as a rich design problem - supporting engaging, emotional, lifelike behaviour design - and as a means to explore the interplay between two design tools, each supporting different activities: *skecthing* and *refining*.

As robots begin to take a larger role in our lives, they require natural ways of interacting with people. Notably, they need to communicate affectively with humans, recognizing and expressing emotion, or behaving with a life-like personality. This is important for both everyday interactions with robots, and targeted health applications: robot-based therapy can measurably relax people by breathing [199].

The Haptic Creature project [**?**] explores the role of touch-based interactions with furry, zoomorphic robots. However, an early prototype of a multi-DoF haptic robot, the CuddleBit, suffers from slow iteration for both hardware form-factor and software behaviours. To explore these concepts more thoroughly, we developed the

(a) "FlexiBit", a furry, flexible CuddleBit.



(b) "RibBit", a rigid CuddleBit.

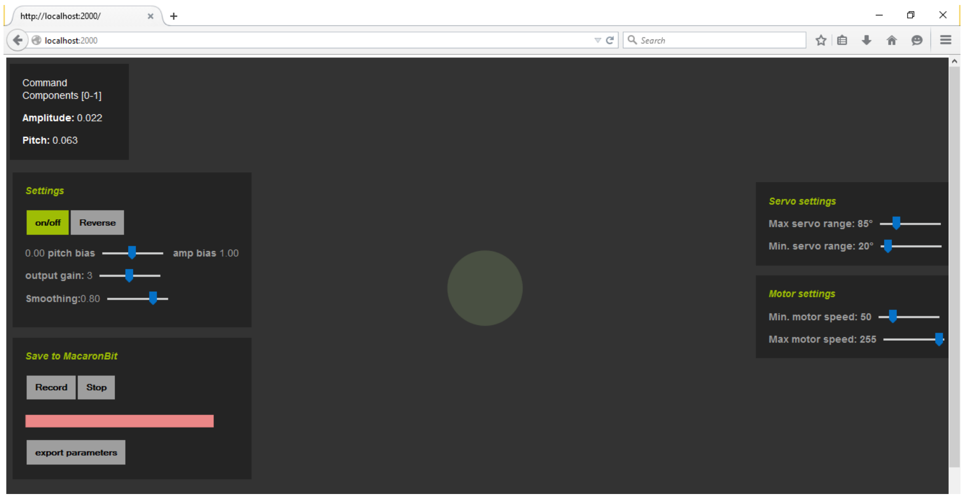
Figure 7.14: Two examples of CuddleBits, simple DIY haptic robots.

CuddleBits [30]: simple, affective robot pals built with a rapid prototyping (sketching) ethic. To control CuddleBit behaviour and inform HaXD support tools in this complex domain, we developed two software design tools: Voodle and MacaronBit (Figure 7.15).

Voodle (Figure 7.15a), from "vocal doodling", is a novel sketching interface to easily create 1-DoF behaviours using non-speech voice, in particular, ideophones [58] like "Ooooh" and "Bwooop." It is inspired by the onomatopoeia descriptions found with our initial exploration (Chapter 3) and previous work [201, 228]. Through a series of user studies, we developed a prioritized set of ideophones and how they mapped to movements of the CuddleBit. At the time of writing, Voodle is in active development; we are using participatory design to further identify critical features, Voodle's expressive capability, and how it might fit into a design tool suite for the CuddleBit alongside Macaron.

MacaronBit (Figure 7.15b) is an adaptation of Macaron (Chapter 5) to control 1-DoF CuddleBit using a familiar track-based metaphor. Instead of two tracks controlling amplitude and frequency, MacaronBit has five: low-frequency amplitude and frequency (for breathing); high-frequency amplitude and frequency (for "shakiness" or "noise"), and bias (asymmetry in the signal), determined during piloting.

Voodle and MacaronBit are symbiotic. Users can record voodles and export them to MacaronBit; initial result suggest that they each support different goals for users. Together, Voodle and MacaronBit represent sketching (Chapter 3) and refining (Chapter 4, showing that this dichotomy provides a useful framing when creating HaXD support tools and applies to other display types beyond VT icon design. Research is ongoing in a series of user studies, exploring expressiveness and consistency of designed behaviours, the specific capabilities and roles of the two tools, and important considerations for future development.



(a) Voodle, a vocal doodling interface that uses voice to control the CuddleBit. The circle in the middle visualizes the CuddleBit's movement on-screen, while additional controls adjust algorithms for vocal processing.

(b) MacaronBit, a version of Macaron (Chapter 5) extended to control CuddleBits.

Figure 7.15: CuddleBit design tools. Voodle enables initial *sketching* of affective robot behaviours, while MacaronBit enables *refining*.

Chapter 8

Haptic Experience Design

Preface – Chapters 3-7 all take a design perspective, investigating by doing. The VT design tools each created short in-lab sessions of design, which we could study directly; however, this approach lacks external validity. Chapter 6 and the focused design projects in Chapter 7 offered ample opportunities to gain implicit knowledge about design; these results are ecologically valid but specific to only one group (our lab). To triangulate both these approaches, we study the wider community: expert haptic designers in the wild. Here, we report findings from six interviews with expert haptic experience designers, augmented by a workshop we coordinated at a major international haptics conference. Chapter 8[1] both grounds our work and serves as a capstone: we define and characterize HaXD as it occurs in practice, codify challenges for HaXD, and develop recommendations to further develop the field that are grounded in this new understanding of designers. We conclude with a vision for how HaXD might manifest in the upcoming years.

8.1 Abstract

From simple vibrations to roles in complex multimodal systems, haptic technology is often a critical, expected component of user experience – one face of the rapid progression towards blended physical-digital interfaces. Haptic experience design is thus now becoming part of many designers' jobs. We can expect it to present unique challenges, and yet we know almost nothing of what it looks like "in the wild" due to the field's youth and the difficulty of accessing practitioners in professional and proprietary environments. In this paper, we analyze interviews with six professional haptic designers to document and articulate haptic experience design, observing designers' goals and processes and finding themes at three levels of

[1]This work has been prepared as a manuscript and is presented as one.

scope: the holistic, multimodal nature of haptic experiences, a map of the collaborative ecosystem, and the cultural contexts of haptics. Our findings are augmented by feedback obtained in a recent design workshop at an international haptics conference. We find that haptic designers follow a familiar design process, but face specific challenges when working with haptics. We capture and summarize these challenges, make concrete recommendations to conquer them, and present a vision for the future of haptic experience design.

8.2 Introduction

Haptic feedback provides value in several ways, especially accessibility [11], low-attention feedback [143], and motor skill training [**?**]. Recently, high-fidelity haptic technology has expanded user experience. Emotional therapy [225**?**], education [185], and entertainment [188] are increasingly employing haptic feedback. Technological advances enable more compelling haptic sensations in consumer products by making it possible to render variable friction on direct-touch surfaces [134**?**], and produce forces without needing to ground devices to a table or wall [**?** **?**]. Even commodity vibrotactile displays are increasing in expressiveness, with high-quality actuation a priority in devices like the Apple Watch (www.apple.com) and the Pebble watch (www.pebble.com), although often at the cost of painstaking and costly design effort. Touch is now increasingly studied within market research because it improves the quality of product opinions and encourages consumer purchases [111]. Part of the power of touch is its emotional, visceral [161] value with it has within a design, giving haptics a close relationship with user experience.

8.2.1 Haptic Experience Design (HaXD)

We define HaXD as:

> *The design (planning, development, and evaluation) of user experiences deliberately connecting interactive technology to one or more perceived senses of touch, possibly as part of a multimodal or multisensory experience.*

Our focus is on gaining a better understanding of the workflow and processes currently used by *hapticians*, the people who practice HaXD as part of their day-to-day work. We describe a two-part, multi-year study that examines how contemporary hapticians design haptic experience for use in real-world products. We begin by identifying current obstacles to good HaXD and the target audience for our work, then provide a roadmap to the rest of the paper.

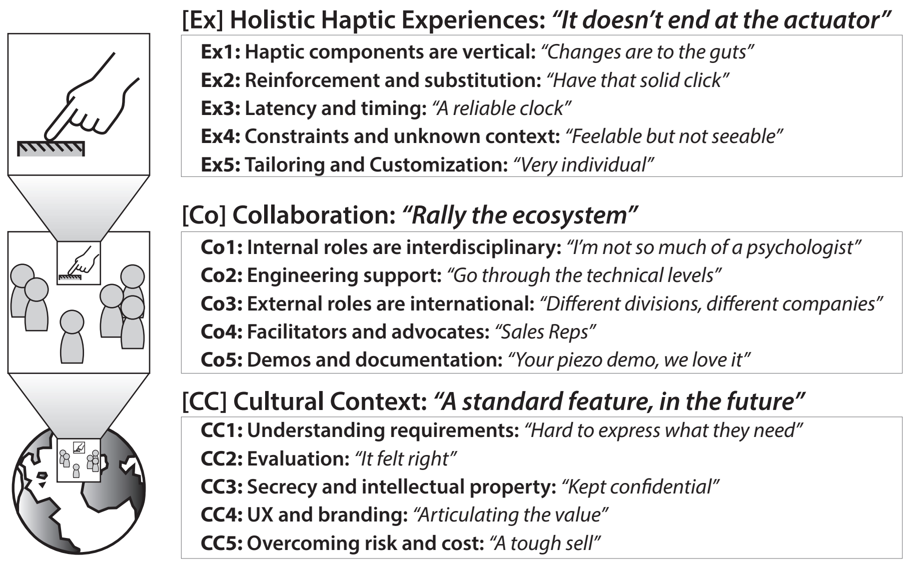


Figure 8.1: Our three themes, each exploring different levels of scope through 5 emergent sub-themes.

8.2.2 Obstacles to Design

The academic literature suggests many challenges to design for haptic experience. Haptic content remains scarce and design knowledge is limited. Some issues are technological, arising in the hardware and software, such as highly variable hardware platforms and communications latency [**?**]. Other issues are human-centered, arising from individual user characteristics in perception and preferences: low-level perceptual variation [138], responses to programmed [134] and natural [**?**] textures, sensory declines due to aging [212, 213], and varied interpretation and appreciation of haptic effects and sensations [201**?**] – often because of personal experience [191].

These research findings are reinforced by many interactions the authors have had with practitioners in industry. Suspecting that there are many challenges related to haptics that face industrial designers, but having little direct evidence to back this up and guide our research, and further suspecting that it is somewhat rare for professionals to design haptic experiences explicitly rather than doing so in the course of larger design efforts, we embarked on a multi-year study of the workflows used by designers when they are engaged in HaXD – something that has been largely unexplored in the literature.

In our study, we take a first in-depth look at haptic designers' experiences

to describe HaXD, identify unique challenges, and connect HaXD to other fields of design. We focus specifically on HaXD instead of the more general notion of "haptic design," which can also refer to design practices related to haptics not directly involving user experience, *e.g.*, mechanical design of a new actuator or software design of a new control method. Our definition encompasses pseudo-haptics [176] and other illusions that trick a user into thinking haptic feedback is occurring without direct tactile or kinesthetic stimulation. Much of what we discuss can also be gainfully applied to the design of tangibles, even with their lack of actuation, although we leave them out of our scope to focus on actuated interfaces. For brevity we sometimes use "haptic designers" to refer to haptic experience designers, while recognizing that beyond these pages this term may include other aspects of haptic design, including purely hardware and software components. We adopt the term "haptician" as shorthand for a practitioner of HaXD.

8.2.3 Target Audience

We primarily target readers who are one step removed from HaXD, but who have other design, haptics, or business expertise relevant to haptics.

We expect that ***haptic experience*** **design experts** (hapticians) will be unsurprised by the insights herein. Although they are not our primary audience, we hope that the articulated challenges and recommendations will nevertheless still be useful for their practice because it consolidates their *ad hoc* knowledge into a formal framework.

We expect that **non-haptic *design* experts** will find our discussion of the specific challenges to HaXD informative because it reveals processes of design that are invisible or are taken for granted in other fields. We also hope non-haptic designers might lend their expertise to accelerate the generation of tools and techniques for creatively working with these complex interactive systems.

We expect that **non-design *haptic* experts** will develop a further appreciation for how UX design is important for successful haptic technology, and will gain an understanding of how their devices or research findings are applied in practice. The recommendations we provide may also motivate several avenues of either basic and applied haptic research that these experts could pursue.

We expect that **industry practitioners** will gain insight into how the business case for haptic technology might be more quickly built. This includes those already involved with haptics or similar technologies such as wearables, as well as those looking to become involved. We believe our findings may help cultivate connections between the diverse stakeholders involved with HaXD, and that the challenges (and thus the opportunities) that we identify will inspire people to work more with this emerging modality.

8.2.4 Roadmap for the Reader

We describe a two-part, multi-year study in which we sought to gain a solid understanding of HaXD as it is currently practiced "in the wild" by actual practitioners (hapticians) in their day-to-day work. After a review of the existing literature in **Section 8.3**, we report on the first part of our study in **Section 8.4**: a grounded theory [44] analysis of intensive interviews with six professional haptic designers. In our results, we describe observations about haptic designers' process organized in three cross-cutting themes: the complex, holistic nature of the experiences they design; the collaborative ecosystem in which haptic experience designers play multiple roles; and the influences of the cultural contexts in which haptic experiences are used and the value and risk this poses. In **Section 8.5** we describe the second part of our study, conducted in a workshop at a major international haptics conference (World Haptics 2015), where we collected quantitative and qualitative feedback from a broader sector of industry and academic designers regarding tool use, collaboration, evaluation methods, and challenges facing haptic designers. In **Section 8.6**, we summarize and discuss our overall findings in three major areas:

1. A description of current HaXD practice showing how it has already emerged as a distinct field of design.
2. A list of challenges facing haptic experience designers, and some unique considerations HaXD requires compared to other more established fields of design.
3. Recommendations for accelerating the development of HaXD as a full-fledged field of design.

We conclude with a few remarks imagining what a mature discipline of HaXD might look like in the near future.

8.3 Related Work

In this section, we discuss key elements of contemporary thinking about user experience design (UX design or XD) and a specific approach known as "design thinking." We then broadly review haptic technology (hardware and software) and relevant aspects of human perception before providing a critical summary of previous efforts to understand and support HaXD.

8.3.1 Design Thinking as a Unifying Framework

Design thinking is an empowering way to approach technology and user experiences. At the heart of this practice is the rapid generation, evaluation and iteration of *multiple* ideas at once [27]. There are several general design activities that we observed in our participants that reflect design thinking, most notably, problem preparation, sketching-like iteration, and collaboration.

Many advocates of design thinking refer to an explicit problem preparation step preceding initial design [198, 205, 227], which involves "getting a handle on the problem" and drawing inspiration from previous work. Designers find value in this stage because creative acts can be accurately seen as recombination of existing ideas, with a twist of novelty or spark of innovation by the individual creator [227]. This stage draws from the designer's experience, including their understanding of the *domain* (symbolic language of the field) [49], and the ability to *frame* a design problem to match it to their repertoire, their their collected professional (and personal) experience [198]. External examples are especially useful for inspiration and aiding initial design [27, 97]. Early and repeated exposure can increase creativity, although late exposure carries a risk of conformity [**?**].

Later in this paper, we describe the evidence we found in our study that haptic designers' work naturally includes a dedicated problem preparation step, *e.g.*, by employing collections of examples in a number of ways.

Sketching is another critical design activity. It supports ideation, iteration, and evaluation. Here, more generally than pen and paper, we refer to general techniques to suggest, explore, propose, and question [27], including physical ideation [155]. Some researchers declare sketching to be the fundamental language of design, much like mathematics is considered the language of scientific thinking [46]. Sketching is rapid and exploits ambiguity, allowing partial views of a proposed design or problem. Detail can be subordinated, allowing a designer to zoom-in, solve a problem, and then abstract it away when returning to a high-level view. It can also support multiple, parallel designs, delaying commitment to a single design [92, 178]. The fluidity and *ad hoc* nature of sketching extends to software tools: designers must be able to rapidly undo, copy and paste, and see a history of progress [178].

We discuss techniques for haptic sketching in prior work to support HaXD, and find major barriers to achieving fluidity that were identified by participants in our study.

Collaboration improves design. Involving more people increases the potential for generating more varied ideas [227], and is recognized as being important for creativity support tools [178, 205]. Although group dynamics can influence the design process negatively, proper group management and sharing of multiple ideas quite often results in more creativity and better designs [97], and can even influence the work of crowds [**?**]. Collaboration can be categorized by intent, such as informal conversations with colleagues or widespread dissemination [205], or by physical and temporal context: collocated (collaborators in the same location) or distributed (in different locations), and synchronous (simultaneous) or asynchronous (at different times) [66].

We find these categorizations useful to identify where collaboration can break down for haptic design, especially remotely, asynchronously, and with limitations on informal or widespread sharing. In Section 8.4.2 we present the first data-informed description of collaboration in HaXD.

8.3.2 Haptic Perception and Technology

Haptic technology is typically separated into two broad classes based on the complementary human sense modalities: *tactile* sensations, perceived through the skin, and *proprioception*, or the sense of body location and forces; the latter includes *kinaesthetic* senses of force and motion. On the human side these are further subdivided into different perceptual mechanisms, each targeted with different actuation techniques. We overview the complexity of the different senses that make up touch, then describe common actuation technologies for these senses, focusing on those mentioned by participants in our study. Finally, we review major application areas that use haptics for both utility and emotional value.

Human perception of touch is synthesized from the tactile and proprioceptive senses, and is influenced by vision and hearing. Tactile sensations rely on multiple sensory organs in the skin, each of which detect different properties, *e.g.*, Merkel disks detect pressure or fine details, Meissner corpuscles detect fast, light sensations (flutter), Ruffini endings detect stretch, and Pacinian corpuscles detect vibration [43]. Proprioception, the sense of force and position, is synthesized from multiple sensors as well: the muscle spindle (embedded in muscles), golgi-tendon organ (GTO) in tendons, and tactile and visual cues [119]. Humans use these senses together to learn about the world, *e.g.*, stroking, bending, poking, and weighing objects in active exploration [126]. Haptic perception is also heavily influenced by other senses. In the classic size-weight illusion [40], when two weights have the same mass but different sizes, the smaller is perceived to be heavier, whether size is seen or felt [94]; similarly, sound can affect how a texture feels [94]. Interactive systems can exploit cross-modal perception to reinforce or improve haptic sensations. To be effective, these effects need to be temporally synchronized, sometimes as closely as 20-100ms [**?**]. For more information about haptic perception, we direct the reader to [43, 119, 127].

Haptic technology to produce stimuli for humans to feel is at least as diverse as the human senses that feel it. Today, the most common approach is vibrotactile (VT) feedback, where vibrations stimulate Pacinian corpuscles in the skin, *e.g.*, smartphone vibrations. VT actuators can take may forms. Eccentric mass motors (sometimes "rumble motors") are found in many mobile devices and game controllers, and are affordable but inexpressive. More expressive mechanisms such as voice coils offer independent control of two degrees of freedom, frequency and amplitude. Piezo actuation is a very responsive technique that is typically more expensive than other vibrotactile technology. Linear resonant actuators (LRAs) shake a mass back and forth to vibrate a handset in an expressive way; a common research example is the Haptuator [233]. Currently, LRAs are increasingly deployed in mobile contexts (*e.g.*, the Apple Watch Taptic engine). Our participants also employ force-feedback, which engages proprioception. Common force-feedback devices include Geomagic Touch (previously the Sensable PHANTOM) and Falcon devices, offering three degrees-of-freedom: force in three directions. At other times, entire screens might push back on the user in a single degree-of-freedom. These are only the most common feedback methods discussed by our participants. Many other types of feedback can be used, *e.g.*, temperature displays [118] or programmable friction display on touch screens [134**?**].

8.3.3 Efforts to establish HaXD as a distinct field of design

Researchers have developed several approaches to support HaXD. Some have directly applied design metaphors from other fields to haptics. Others have built collections of haptic sensations and toolkits that facilitate programming. A number of haptic editors, analogous to graphical editors like Adobe Illustrator, have emerged to support specific haptic modalities through parameterized models or other abstractions. These approaches have developed focused understandings of particular aspects of HaXD, but they do not adequately describe the process as it is actually practiced.

There are many examples of designers drawing from other fields to frame the practice of haptic design. Haptic Cinematography [57] uses a film-making metaphor, discussing physical effects using cinematographic concepts and establishing principles for editing based on cinematic editing [86]. Similarly, Tactile Movies [123] and Tactile Animation [196] draw from other audio-visual experiences, and Cutaneous Grooves [88] draws from music to explore "haptic concerts" and composition as metaphors. Academic courses on haptics are taught with a variety of foci, including perception, control, and design that provide students with an initial repertoire of pre-existing skills drawn from other disciplines [114, 167]. These and other ways of framing HaXD have been incorporated into rapid prototyping techniques that allow for faster, easier iteration of haptic designs. Simple Haptics, epitomized by *haptic sketching*, emphasizes rapid, hands-on exploration of a creative space [155, 156]. Hardware platforms such as Arduino (arduino.cc) and Phidgets (phidgets.com) [83], as well as the recent trend of DIY haptic devices [74, 77, 168], encourage hackers and makers to include haptics in their designs.

The language associated with tactile perception (terms related to haptic sensation and how they are used), especially affective (emotional) terms, is another way of framing haptic design. Many psychophysical studies have been conducted to determine the main tactile dimensions with both synthetic haptics and real-world materials [67, 164]. Language is a promising way of capturing user experience [162], and can reveal useful parameters, *e.g.*, how pressure influences affect [242]. Tools for customization by end-users, rather than by expert designers, are another place that efforts have been made to understand perceptual dimensions using a language-based approach [200, 201]. However, this work is far from complete; touch is difficult to describe, and some researchers even question the existence of a tactile language [111].

Meanwhile, software developers who want to incorporate haptics into their systems are supported by large collections of haptic sensations and programming toolkits. Sensation collections most commonly support VT stimuli. The UPenn Texture Toolkit contains 100 texture models created from recorded data, rendered through VT actuators and impedance-type force feedback devices [51]. The Feel Effect library [108], implemented in FeelCraft [188], lets programmers control sensations using semantic parameters, *e.g.*, "heartbeat intensity." Immersion's Haptic SDK (immersion.com) connects to mobile applications, augmenting Android's native vibration library with both a library of presets, and on some mobile devices, low-level drivers for effects like fade-ins. VibViz [201] is a free on-line tool with 120 vibrations organized around five different perceptual facets. Force-feedback environments tend to be supported through programming toolkits. CHAI3D (chai3d.org), H3D (h3dapi.org), and OpenHaptics (geomagic.com) are major efforts to simplify force-rendering. Table-top haptic pucks can use the HapticTouch Toolkit [128], which includes parametric adjustment (*e.g.*, "softness") and programming support.

Finally, several software-based editing tools support haptic design for different devices. These tend to focus on VT stimuli or simple 1-degree-of-freedom force feedback. Many editors [67, 184, 194, 215**?** , 216] use graphical mathematical representations to edit either waveforms or profiles of dynamic parameters (torque, frequency, friction) over time. Of these, Vivitouch Studio [216] offers the most integration with other modalities in games, and Macaron [194] is the most available tool (online and web-based). The Vibrotactile Score [132] uses a musical metaphor, shown to be preferable to a programming metaphor as long as the designer has musical experience [130]. Mobile "sketching" tools like the Demonstration-Based Editor [99] and mHIVE, a Haptic Instrument [191] are useful for exploration, but not refinement. Since iOS 5 (2011), Apple has let end-users create on/off vibrations as custom vibration ringtones. Immersion's Touch Effects Studio lets users enhance a video from a library of tactile icons supplied on a mobile platform. Actuator sequencing [169], movie editing [123], and animation [196] metaphors enable multi-actuator, spatio-temporal VT editing.

Some of these tools are founded in an understanding of haptic designers' needs [196, 216] and begin to capture a slice of the HaXD process [194], but they do not fully capture the context and activities of contemporary haptic design.

8.4 Part I: Interviews with Hapticians about HaXD in the Wild

In this section, we present findings from the first part of our study, a qualitative analysis of interviews with six professional haptic designers who we refer to as "hapticians."

8.4.1 Method

One researcher (the first author) analyzed the interview transcripts through grounded theory [44] influenced by phenomenology [157] and thematic analysis [183]. The analyzing author, who was trained in qualitative methods, first transcribed interviews and then examined every participant statement, tagging each with relevant and recurring concepts and keeping written notes for reflection and constant comparison. Emergent sub-themes (sub-categories) [183] were discovered using qualitative techniques of memoing, iterative coding [44], clustering and affinity diagrams [157]. Statements were later grouped according to tags, organized using affinity diagrams and clustering, and iteratively developed with further writing and reflection. The 15 sub-themes clustered into three themes (categories) [44, 183]. We describe the themes in Section 8.4.2 after an introduction to the designers themselves and the procedure that was followed for the interviews. We delay a detailed discussion of the results until Section 8.6 so we can include the findings of the second part of the study, presented in Section 8.5.

Participants

Six participants were recruited, 5 male and 1 female. Initial contact was by email from a list of potential interviewees developed by the researchers. We describe each in terms of experience and training, area of focus within HaXD, types of projects, and constraints or other factors that might situate or provide insight into the interview. Experience and position are reported as of the interview year (2012).

P1 (M, over 15 years of human factors experience, PhD) held a design and human factors position at major healthcare company. He worked with auditory alarms, signals, and emotional experience. Despite a focus on audio, he frequently related his work to haptics and works with physical controls, designing characteristics like force profiles and detents, and described the haptic and audio processes as being the same. Working in health care means there are tight regulations that need to be followed, and a noisy, diverse environment. P1 used a number of psychology and human factors techniques, such as semantic differential scales, factor analysis, and capturing meaning.

P2 (M, 5-6 years in haptics, PhD) described two projects: his experience adding mechanical feedback to touch screens at a major automotive company, and his PhD work on remote tactile feedback, where feedback was displayed on one hand while the other interacted with a touch screen. P2's main concern is "rich feedback", communicating information like affordances to the user. This is both pragmatic, such as "consequences" of the button, and affective, aiming to have sensations "feel right." P2 focused on button presses on a touchscreen, rather than exploring "roughness" of a touchscreen or other surface.

P3 (M, 10 years leadership experience with actuation, sensing, and multimedia, M.Eng.) worked at a company that sells actuators used to add haptics to technology (like a tablet computer, game controller, or mobile phone). P3 had 20-30 projects going on at any time, each with their own level of size, goals, constraints, and other contexts. His main goal was to sell a developed actuator (with several variants).

P4 (M, 11 years of design, development, and analysis/simulation experience, PhD) also puts actuators into new form factors (*e.g.*, touch screens in cars). When he worked, he had limited time and resources, so there is not much time to change things.

P5 (M, 12 years of haptics UX experience, M.Sc.) held a user experience leadership position at major haptics company that sells haptic control technology and content; he described mostly software solutions. His company worked with different domains, but most examples are from mobile phones (handhelds), with a brief mention of automotive haptic feedback. They worked with extremely high-end piezo vibration actuators with high bandwidth (frequency and mechanical), and delivered software solutions for Android to their customers: OEMs (original equipment manufacturers). He described handheld feedback as two different classes: confirmation haptics, like a vibration to indicate a widget has been used, and animations/gestural feedback, which is more complicated.

P6 (F, 5-6 years in haptics, PhD) worked at a major car manufacturor. She primarily designed "feel" properties such as friction, inertia, and detents of physical controls inside automobiles. P6 also works on active haptic controls. Design aspects include measuring force vs displacement profiles and maintenance of a large scale haptic design specification repository that spans user and technical requirements. This haptic specification repository is used by many engineering and business stakeholders across many sites in different countries.

Procedure

One co-author interviewed the 6 participants in April-May 2012 using Skype. Each interview lasted 30-60 minutes and consisted of initial ice-breaker and general open-ended questions, then increasingly detailed questions about participants' processes. Interviews followed a semi-formal, structured qualitative inquiry process that allowed for followup. Interviews with P2-P5 were fully recorded and transcribed. Interviews with P1 and P6 were collected only as interviewer notes. In the presentation of our findings, double quotation marks ("...") denote direct transcription quotes for P2-P5 while single quotation marks ('...') denote interviewer notes for P1 and P6.

8.4.2 Results

Most of the emergent themes that we identified persist throughout the design process (Figure 8.1). We found participants generally followed a process typical of experience design (UX) [27] in which they first tried to gain an understanding of the design problem, then iteratively developed ideas and evaluated them. We first outline these confirmatory observations on process, then go on to report on the themes, our main findings. Throughout, we cross-reference themes by section number and theme label (*e.g.*, 8.4.2/Co5).

Observations on Design Process

Participants described the initial stages of a project as a time to establish and understand requirements, gather initial design concepts, and define or negotiate project parameters. Designers often collected examples of haptics, such as mechanical buttons and knobs, for inspiration (8.4.2/Co5), and they gathered requirements – both direct requirements for haptic designs (8.4.2/CC1), and project parameters around the value, cost, and risk of haptic technology (8.4.2/CC4,CC5).

P2-P6 explicitly referred to an iterative process. They all found different ways to fit it into their collaborative ecosystem and constraints. As we elaborate below, prototyping and assessment in the physical medium of haptics has many challenges that set it apart from graphical or auditory domains even as designers navigate very common-place objectives. For example, initial requirements were often not actually what clients wanted, and our designers would have to iterate (8.4.2/CC1). P5's teams explicitly follow a conventional user-centered design process, iterating simultaneously on prototypes and their understanding of customer needs. P3 sometimes has to ship mockups and devices back and forth with their customers (8.4.2/Co5). Each design problem faced by our participants had to be treated as a unique problem, with designers fine-tuning their design to fit the problem

(8.4.2/Ex5). Our designers used a variety of evaluation techniques to choose their final designs (8.4.2/CC2).

We now proceed with our cross-cutting themes, organized by scope (Figure 8.1): the haptic experience and its implementation (Section 8.4.2), the designers' collaborative ecosystem (Section 8.4.2), and implications from the wider cultural context of haptic technology and business requirements (Section 8.4.2).

[Theme Ex] Holistic Haptic Experiences: "It doesn't end at the actuator"

Context is crucial to experience at multiple levels, but is difficult for a designer to foresee or control. Aspects of context range from immediate, very local electromechanical environment (material properties, casing resonance, computational latencies), through the user's manner of touching the haptic element (grip, forces, longevity of contact), to the user's momentary environment, attention, and goals.

At the local end, the complexity of the haptic sense itself is a major factor in expanding the haptic experience design space substantially beyond what are usually its initial requirements – for example, for the changing feel of a modal physical control in an automobile cockpit. As we've discussed, the haptic sense is really a collection of subsenses [43, 119], working together to construct an overall percept, *e.g.*, material properties deduced from stroking, tapping, or flexing a surface or object [126]. Grip, materials, dynamics as well as visual and audio aspects all play a part in the result.

> *"The problem is it doesn't end at the actuator, there's a lot to do with the case of the device, the mass of the device, the mechanical coupling between the device and the hand...this all comes into play because it's a tangible experience, and so if there's mechanical resonances that get stimulated by the actuator that make it sound noisy, then it becomes a cheap experience, even if it has a piezo actuator."* (P5)

Thus, designers both face multifaceted constraints and have opportunities to circumvent those constraints. We begin by discussing implications for implementation, wherein haptic components are directly related to the internal mechanics – the "guts" – of the system (Sub-theme Ex1). Then, we move on to opportunities for improving design: strategies like reinforcement and substitution are powerful tools for haptic designers (Ex2). Timing is critical, enabling abovementioned opportunities while imposing constraints: designers must introduce no new delays and carefully synchronize feedback (Ex3). However, the full extent of a sensory context is sometimes uncontrollable or unknown, and at such times prevent designers from using their tricks (Ex4). We finish this section by discussing how haptic experiences are often bespoke, tailored to constraints of known contexts, or customizable to unknown contexts (Ex5).

Table 8.1: Sub-theme summaries for the Holistic Haptic Experiences (Ex1) theme.

Code	Sub-theme descriptor	Explanation
Ex1	Haptic components are vertical	Changing a haptic component may influence the larger hardware/software system, and vice-versa.
Ex2	Tricks to create great feels	Haptic designers can improve designs and work around constraints through multimodal tricks.
Ex3	Latency and Timing	Without fast feedback and synchronized timing, haptic experiences fall apart.
Ex4	Constraints and unknown context	Other modalities may impose constraints; constraints may not always be knowable.
Ex5	Tailoring and customization	Designers tailor their solutions to each application; end-users benefit from customization.

Ex1: *"Changes are to the guts"* – Haptic components are vertical. Haptic experiences are created when the actuating component physically interacts with other system components. Changing a haptic component can thus affect the entire system's design, unlike many other upgrades, like improving memory in a mobile phone: *"you get the impression every other month they have a new phone...but the guts of it do not change much"* (P3). New phones often just have a faster CPU or more memory swapped into an essentially unchanged system; but when adding or modifying haptic components, designers must consider the entire system including the physical casing, and possibly modify it as well:

> *"First we had to get the outer dimensions [of the prototype's case] roughly about right, to get the visual impression close to what it resembles later in the application"* (P4).

This effect is bidirectional. Changing the size or material of the casing can have a profound effect on the sensation; correspondingly, any changes to the haptics will have an effect on the entire structure of the device. Changes to software are also cross-cutting: *"we're digging into the source code of Android...we need to make sure that we have the right hooks in the right locations...that's a software architecture issue, right?"* (P5).

Ex2: *"Have that solid click"* – Tricks to create great feels.
Haptic designers have an array of techniques to create great experiences, working around constraints and uncertainty. The first step is to have a fast, responsive actuator when possible. Previously, creating good actuators was a goal for our participants: *"[what we] strived in the past significantly to do was to push the market towards high mechanical bandwidth actuators, so actuators that can respond in 15 milliseconds or less"* (P5). Now, high-quality actuators are a main competitive advantage:

> *"High-definition feels over a very broad frequency range, with enough strength and small enough, and especially very fast response time, that's our business"* (P3).

As discussed in Ex1, the actuator does not determine the experience alone, but interacts with physical materials and non-haptic senses. When a haptic device's ultimate situation is known at design time – like a car dashboard – designers can modify properties of the larger physical system to improve the overall haptic experience: *metal makes unwanted sound, so change it with a plastic* (P6). The designer can also make a sensation more convincing with multimodal reinforcement, *e.g.*, adding visual or audio feedback:

> *"Need to have that solid [haptic] click at 150 [Hz] plus some audio at 300 or 400 Hz, which is going to give you that sense of quality, and, consistency across the whole dashboard"* (P5).

When a known physical context has constraints, designers also use substitution to enable or improve the haptic interaction. P2 describes two such occasions, one for sensing input and one for displaying output. Because P2 could not sense input pressure, he instead used how long the user was pressing the screen ("dwell time"): *"we were substituting the forces that are needed on the actual buttons with dynamic dwell times"* (P2). This was only possible because P2 had knowledge of *how* the user would be touching the control, and thus could deduce that dwell time was a reasonable proxy for pressure. In another case, P2 could not actuate a touch screen, so he used tactile feedback on the other hand – again, requiring knowledge of and considerable design access to the device's and user's larger situation; here, the steering wheel.

Ex3: *"A reliable clock"* – Latency and Timing.
One underlying requirement for great haptic experiences is responsive timing. Feedback must be fast; modalities must be synchronized. Effective reinforcement requires simultaneity and hence tight (millisecond) control over timing. This is well

established in the literature [135?] and known to our designers: *"I think, audio feedback and tactile feedback and visual feedback has to happen at a certain time to have a real effect"* (P2).

Latency accumulates throughout the computational pipeline, with actuator responsiveness the very last stage and rarely the most impactful. Designers must minimize computational delays wherever possible. P2 describes unintentionally adding latency to one project: *"we had this Python program and Arduino and all this communication going on"* and how he *"threw out some of the serial communication which [had] made the whole thing a little slow"*, and thus, the *"latency again felt right"*. Timing problems between components can happen at any time: *"we've gotten in situations before where we've been very near to completion in design projects, and for whatever reason we can't get a reliable clock, from the CPU, then the whole thing falls apart"* (P5).

When adequate simultaneity constraints are met, the user perceptually *fuses* these non-collated events (activating a graphical element on a screen, and feeling a tick on the steering wheel) into a single percept: *"somehow you connect these two things, the action with the dominant hand and the reaction that is happening somewhere else"* (P2). Haptic designers thus need access to the computational pipeline to circumvent physical constraints with multimodal tricks.

Ex4: *"Feelable but not seeable"* – Unknown user constraints and context. Haptic designers sometimes contend with unavoidable constraints emerging from physical context or application space. Some constraints not only limit multimodal synergies, but go on to actively limit haptic display. For example, eyes-free interaction in cars means that visual reinforcement is unavailable; indeed, visual movement may have to be avoided altogether for safety reasons. P4 is tasked with creating a *"feelable but not seeable"* sensation to *"avoid having to use visual feedback"*, as *"driver distraction is always a big topic"* (P4). This means P4 has limited control over his designed haptic sensation, as it cannot visibly move, but P4 can use audio reinforcement or substitution to handle constraints.

Perhaps even more difficult is when the experience's context is unknown. This can derive from at least two sources: protection of intellectual property (IP) through secrecy, and unconstrained end-user situation. Stakeholders often keep key contextual information such as the visual interface secret from third party designers (*e.g.*, OEMs [original equipment manufacturers] or consulting): *"we can suggest components, and suggest characteristics of the HMI [human-machine interface] system, but the exact visual design of the HMI system is the OEM's knowledge"* (P4). P3 has an *evaluation kit* to send to potential customers when customers' IP is a concern:

> *"[An evaluation kit is] basically a little box that consists of our actua-*

> *tor and some electronics, and that box is connected and driven through the USB port of a computer, and you can then mechanically integrate the box in your own way, so we don't need to know what their design looks like"* (P3).

We discuss IP and secrecy more in Section 8.4.2/CC3. Meanwhile, designers must deal with sometimes unknowable end-user context, especially with mobile scenarios. A high quality LRA-type actuator on a metal table can sound cheap, while an affordable eccentric mass actuator can sound like purring if it's on rubber, and *"there's not much you can do from a haptic perspective, other than allow the user to turn it up or down"* (P5).

Ex5: *"Very individual"* – Tailoring and customization. Because the context of haptic technology can vary so much, haptic designs need to be tailored for each client's problem and are often made customizable for end-users. For the former, several participants' business models are directly based on tailoring. P4's group makes a small set of actuators, adapting them to each specific request. This is exacerbated because it is *"hard for [customers] to really express what they need"* (P4) (discussed more in Section 8.4.2/CC1) so designers must rapidly and collaboratively fine-tune their solutions.

Even if customer goals are clear, tailoring is necessary because of requirements (*e.g.*, branding or *"trademark"* (P2), 8.4.2/CC4) and hardware setup: *"it's important to tune the experience depending on whatever kind of motor they decide to put in"* (P5).

> *"Depending on the outer design, what's given to us by the customer, we have to choose the direction of movement. For some applications, for some ideas, it's possible to move the surface directly perpendicular, away from the user, and other applications, you have to move the surface perpendicular towards the user, so the same actuation module could feel completely different"* (P4).

Meanwhile, individual differences of end-users further complicate matters: *"feeling right is...something that is very individual"* (P2). As P5 mentioned, volume controls can help end-users and adapt to unknowable context.

[Theme Co] Collaboration: "Rally the ecosystem"

In this section, we describe the collaborative ecosystem. First, we provide an overview of group structure and interdisciplinary roles found in our participants' groups (Sub-theme Co1), including a focus on the role of engineering (Co2). We

Table 8.2: Sub-theme summaries for the Collaboration (Co) theme.

Code	Sub-theme descriptor	Explanation
Co1	Internal roles are interdisciplinary	It takes a multidisciplinary team to create a haptic design.
Co2	Engineering support	Prototyping is necessary and often delegated to engineers.
Co3	External roles are international	Haptic design teams work with other stakeholders around the world.
Co4	Facillitators and advocates	Sales reps handle demos and fight for a deal.
Co5	Demos and documentation	Designers often show instead of telling.

Table 8.3: Internal roles, the various descriptors used to label them, and descriptions. Roles were grouped and named by the authors based on participant-provided descriptors.

Role	Descriptors	Description
UX	User division (P6), User research (P5), Ergonomics (P6), Human factors (P1), Psychologist (P2,6)	The UX team does research: *facilitate prototypes, validate, communicate those results* (P5). Here we include psychologists and human factors roles because they conduct user research such as evaluation: *psychologists there who do usability tests* (P6), *study how effectively how users interact w/ goals* (P1).
Design	Design team (P5)	Related to UX but a separate and in some ways higher-level role. The design group ideates and communicates vision, developing a value proposition. Designers usually have a similar background to the UX group (P5).
Engineering	Tech manager (P3), Engineering (P3,5) Electronics, mechanics, tech team (P6)	Often a separate division, handling prototyping and implementation (P5). They might test components, do physical construction, take requirements from design, ergonomics, electronics, mechanics, etc. and generate required (haptic) feedback (P6). This can involve both hardware and software.

then discuss the dispersion of stakeholders internationally and in different organizations (Co3), including a focus on the connecting role of sales representatives, and the use of demos and documentation (Co5). We distinguish the Collaboration theme (Section 8.4.2) from the Cultural Context theme (Section 8.4.2) by focusing on specific communication methods and roles rather than underlying values and widespread public consciousness.

All six participants indicated collaboration was an important part of their work and design process. Haptic designers are part of interdisciplinary, international teams, and do not make haptic experiences alone:

> *"We basically have to rally the ecosystem...we have to go and find, y'know, somebody to supply the amplifier part, somebody to make the motor, somebody who knows enough about the Android kernel...we have to be, kind of, renaissance men if you like"* (P5).

Co1: *"I'm not so much of a psychologist"* – Internal roles are interdisciplinary. Haptic design is interdisciplinary; hardware, software, psychology, and business all play a role. P5 describes his company's job as *"rallying the ecosystem"*, finding diverse expertise and establishing a production chain. P6 describes different roles in her team, who work more closely together at different stages: *user [research], design, ergonomics, haptics, electronics, all come together* (P6). This is reflected by the diverse internal roles (Table 8.3), but also in the diverse work in single projects and individuals:

> *"We do some mechanical integration work, we help [our international customers] with designing the electronics, we have reference designs there, we have a couple of reference effects, and then we ship the part back and they go on with further doing the software integration and designing the haptic effects."* (P3)

Our participants worked in groups of various sizes. P2 worked with a student in a team of 2, while P5 describes several teams: design, UX, engineering, each with different responsibilities. This collaboration can be collocated or remote: P6 describes the different divisions in her company as being physically close together, while P3 has sales representatives (*"reps"*) overseas to help with external collaboration.

Especially in smaller groups, team members fill multiple roles. Sometimes this falls naturally into their background: *"I guess [phone vibrations are] similar to mechanical control design, except that it's all virtual"* (P5). Otherwise, this lack of expertise leads reduces confidence: *"I don't know, I'm not so much of a psychologist to really, to dare to say I can evaluate subjective responses to tactile feedback"* (P2).

Co2: *"Go through the technical levels"* – Engineering support. Larger groups are able to have more specialized individuals. Especially common was a dedicated engineering or technical support team, tasked with implementing prototypes for design and user research teams.

> *"In our design research team we don't do any internal prototyping, we rely on engineering resources to do all our prototyping"* (P5).

P5's group says that neither the design team nor the UX team build prototypes, though the UX team facillitates and evaluates them. P1's team is similar: *give qualitative feedback and ranges to the technicians* (P1). Engineering departments are sometimes *physically very close to other departments* (P6), presumably to interact with different divisions and groups. However, separating expertise can cause gulfs of collaboration, *e.g.*, when P3 tries to propose a deal:

> *"If you try to go through the technical levels from a technology scout to a technical manager and then maybe to a senior manager, you usually get blocked with something, because nobody wants to take the risk or the blame"* (P3).

Those in engineering roles are risk-adverse: *"[it's] risky to suggest changes to their component"* (P3). P3 says that to pitch to other companies, you need to reach *"C-level people"* like the CEO, or other business or manager types: *"engineers look at it from a perspective well I'm going to take a risk if I change something in my design, and if it doesn't work everybody's going to blame me"* (P3), *technicians won't give pushback if there is a problem* (P1).

Co3: *"Different divisions, different companies"* – External roles are international. Haptic designers also worked closely with external stakeholders like potential customers and manufacturers. Our designers have diverse suppliers, especially hardware suppliers, and often sell to manufacturers who then sell their own product to the end-user. Table 8.4 provides details on these external-facing roles.

> *"Automotive is very much a tiered and compartmentalized manufacturing business, and so the person who makes the control surface is different than the person who makes the mounting for it...and those people often never talk to each other, and so for us it's even worse than different divisions in a company, it's different companies"* (P5).

Often these groups are distributed internationally. P5's group, based in North America, received international demographics to research: *"here's phone X from OEM Y and it's targeted at Asian ladies from 15 to 30 years old"* (P5). P3, who has a headquarters in the North America and clients in Asia, describes *sales reps* as critical team members who can bridge language and cultural barriers.

Co4: *"Sales reps"* – Facilitators and advocates. P3 describes sales reps in-depth as key team members. Sales reps are trained locally at headquarters in North America, then are sent to the customers' area, often in countries like Korea, Japan, China,

Table 8.4: External roles, the various descriptors used to label them, and descriptions.

Role	Descriptors	Description
Connections	Sales rep, technology scout (P3)	Sales reps from haptic companies handle local expertise (language and culture), haptics expertise (they run demos), and can be advocates for products. Technology scouts from large companies talk to haptics companies to learn their technology.
Business	Business dev people, C-level people (P3)	Internal business development people are *"here [in HQ]"* (P3), while external business people make decisions; they're who you need to persuade, rather than technology-focused roles.
Supply chain	Vendor, developer, manufacturer, OEM (P5), supplier (P4,6), content provider (P3)	Haptic designers are heavily embedded in a supply chain involving hardware and software manufacturers. Some manufacturers provide hardware (e.g, actuators) and software (e.g., Android API) to the haptician, others are the intended customer (phone or car manufacturers, software developers). It is unclear who creates haptic content in this ecosystem.

and Taiwan which have large consumer electronics and gaming markets. It is important that they speak the local language and understand the local culture; they also facilitate demos and persuade customers to pursue business with the designer's team. If a demo is sent to a company without a sales rep, customers may respond by shipping the device back and requesting assistance, but often don't respond at all:

> *"If we try to just ship them a part...in the best case they come back and say well it doesn't work as we thought, can you help us?...in the worst case they don't even contact us back and we never learn why they didn't pursue an idea or an opportunity. It's still a complicated setup to make haptics work, there's lots of aspects that you have to take into account, and if you don't do it properly, you're going to be most likely very disappointed about what the outcome is"* (P3).

Big tech companies sometimes invert this from a push model (where the haptics company uses a sales rep) to a pull model with tech scouts (who reach out to haptics companies). Sometimes, companies fill this role without dedicated sales reps: P4 goes to customers regularly in confidential meetings, receiving specifications and working collocated with the customer to get their product to feel *"just right"*:

> *"There is always [the] option, as we did with one of our customers, that we simply went into the lab for a day or two, and just worked on simulated button feel, together with the customer, to get the feel just right"* (P4)

In all cases, content can fall through the cracks. P3's company provides technology, but *"the issue that we are having with uh, the content providers that need to get interested and believe in it...creating the haptic effects is something that we haven't been involved in in a lot of detail in the past"* (P3). P5's company does have a set of 150 effects, from which they select themes. The other participants all mention technology they develop, with content directly related to their hardware solution.

Co5: *"Your piezo demo, we love it"* – Demos and documentation. Demos are essential to showing both the value of a haptic experience and enabling two-way communication with the customer. They can clarify requirements and grab attention from clients: *"we'll often get the OEMs who will say, well you showed us your piezo demo, and we love it, it feels great"* (P5). Demos can be conducted in-person (synchronously) at events like tech-days or one-on-one meetings: *"the customer either comes directly to us, we go towards our customers regularly, have our tech days, similar to automotive clinics"* (P4), or asynchronously, remotely shipped.

However, demos are complicated and need an experienced handler like a sales rep. Once set up, demos are often adjusted, but this is easier than the setup: *"From the moment the actuation module was working...it was just cranking up the maximum current or reducing the maximum current"* (P4).

Demos are often collected into groups. P5 describes downloading apps that use his technology and *"sticking those in [their] demo suite"*. P1 and P2 talk about collecting examples for inspiration and guidance early in design: it's *quicker to go out and buy examples*, like *15 or 16 appliances that had notably different feelings* (P1). P2 instructed his student to *"collect physical push buttons just to get in contact with all the diversity of stuff"*, and ended up with a *"button board"* to guide design. He also talks about company guidelines:

> *"When I was at [a major automotive company] 3 years ago...they had this guideline book...they had guidelines on the design of physical widgets like sliders, physical sliders, push buttons, rotary things...they defined thresholds basically where these forces have certain thresholds and if you get over the threshold something is happening"* (P2).

Demo setups can thus be stored long term for internal documentation (button board, guideline book), but they can also be ephemeral (tech days). In both cases, they can help to articulate the value, especially valuable when most people do not yet understand haptic technology.

[Theme CC] Cultural Context: "A standard feature, in the future"

Haptic technology has yet to fully penetrate the public consciousness. Participants reported major difficulty when working with both customers and users, including a limited understanding of what haptic technology is and how to work with it:

Table 8.5: Sub-theme summaries for the Cultural Context (CC) theme.

Code	Sub-theme descriptor	Explanation
CC1	Understanding requirements	Customers and designers have trouble articulating and understanding goals.
CC2	Evaluation	Getting experiences to feel right, usually with acceptance testing and deployment.
CC3	Secrecy and intellectual property	Haptic technology and sourced components are often cutting edge and secret.
CC4	UX and branding	Tactile experiences provide intangible benefits.
CC5	Overcoming risk and cost	Haptics are risky and expensive to include in a product.

> *"People really don't know what to do with [haptics] and I think within the haptics community we need to...continue to push it into the market, but once it's there I think it's going to add to the user experience and will be a standard feature in the future"* (P3).

Specifically mentioned were the difficulty in understanding customer requirements (CC1), and knowing how to appropriately evaluate haptic experiences (CC2). As a technological field, secrecy and intellectual property are important concerns for both designers and customers (CC3). Designers had ways to pitch the value proposition of haptics, often tied to UX and branding (CC4), but risk and cost of adopting the technology often make it a hard sell (CC5).

CC1: *"Hard to express what they need"* – Understanding requirements. Customers found it difficult to both understand and request their needs. Our participants focused on the end result because it gives them and their colleagues the ability to solve problems: *Don't specify elements. Only give end product. Don't tell how to restrict; can give hints* (P6). However, requested end-results are often vague or confusing, like *"good variable feel"* (P4):

"The customer only came with a question, yeah, how [can the design] feel variable? Here it did not really describe how it should feel variable" (P4).

To make these impressions concrete, customers initially give engineering parameters as their best guess. P4 in particular talks about his customers, who might point to a *"reference button which is available directly on the market, from companies like [company 1] or [company 2], and they say it has to feel exactly like this button"*, or request *"a surface acceleration of 10 to 20 G perpendicular and a travelling distance of .2-.3 mm"* (P4). This might have little relation to the final result, after the designers iterate with the customer: *"we ended with an acceleration of 2G and a travelling distance of .4 of a mm, so, due to the size of the module, simply the high accelerations were too high for a good variable feel"* (P4). The goal function of good variable feel was achieved, but the initial engineering-level specification was completely off.

Other participants showed this duality between high-level affective goals and low-level guesses. P1 especially used affective and psychological terms when considering design, such as semantic differential scales: *good/bad; gender (robust/delicate; size); intensity (sharp/dull; bright/dim, fast/slow); novelty* (P1). Haptic designers often connected low-level/high-level terms through iteration, or with their own way of representing features like quality: *"[audio click gives] quality, and, consistency across the whole dashboard"* (P5), *mass is big for quality...for the haptics, nice feedback w/ good snap gives a sense of quality* (P6).

CC2: *"It feels right"* – Evaluation. Our designers all evaluated their designs but demonstrated different methods of evaluation, consistent with our workshop survey (Section 8.5). P2 explicitly evaluates both low-level, pragmatic concerns (*e.g.*, task accuracy and speed) and high-level affective concerns like feeling personally involved (with the AttrakDiff questionnaire [**?**], http://attrakdiff.de). P5's user experience team conducts validation (but was unable to share details). Small-scale acceptance testing was employed by both P2 and P4: when iterating in-person with the customer, P4 kept iterating until the customer said it *"felt right"*; P2 only had himself and his student evaluate their designs in an academic context, despite indicating a desire to do a more thorough evaluation. P3's group doesn't create content, but indicated a desire to look into that and investigate it with studies.

Our participants expressed a clear desire for stronger evaluation, but reported mostly lightweight, *ad-hoc* acceptance testing. This is consistent with our workshop findings, which suggest little real-world or *in situ* evaluation. One reason may be that evaluation tools need to be adapted. P2 describes having to *"throw out"* terms on the AttrakDiff questionnaire that did not fit, and iterate on the questionnaire. However, deployment seems to be a natural way to see if the design is good enough, as the ultimate acceptance test. P5 described the most memorable moment of his software project being when his product had been deployed and used by a software development team. Seeing a haptic-enabled app available for download, and feeling it in context, was impressive:

> *"I think the most memorable day was when we started downloading apps, and realized that, yes, in fact this does work, and not only does it work but it works pretty well for a variety of apps... we ended up just sticking those in our demo suite even though we had no relationship whatsoever to the developer. So, their app just worked, and it worked really well"* (P5)

CC3: *"Kept confidential"* – Secrecy and intellectual property. Sometimes the customer does not know what they want, but in other cases, they have important information they need to withhold. As mentioned in Section 8.4.2/Ex4, secrecy in haptics has major implications that inhibit design, especially given the verticality of haptic technology:

> *"Somebody wants to design a completely new gaming controller for a gaming console, so they might just have some CAD drawings or they might have something they don't want to share with us, so in that case we provide them an evaluation kit...we don't need to know what their design looks like, they can really work on it internally"* (P3)

P3's clients are able to receive an *evaluation kit* and create content with audio editors. P4 describes meetings with customers that preserve confidentiality: *"on these tech days it's usually only one customer and not that many suppliers at the same time, sometimes only the customer and us, to make sure our development is kept confidential"* (P4). Once technology of P4's company is on the market, it is no longer secret - rivals can copy or reverse-engineer the devices, so there are many demonstrations to customers before release of the tech. P4 wants to show their technology to potential buyers, not competitors.

Secrecy can cause delays for software too. P5 delivers a modified Android kernel to his customers, who are software developers. However, they aren't given an early release, and thus they *"always lag the market by two months at least, to get an update [for Android]...it's annoying because, you know, as soon as the OEMs get the source code they want to put it in their product right away"* (P5).

CC4: *"Articulating the value"* – UX and branding. Our participants were all passionate about haptic technology and its benefits. The value of haptics can be connected to better performance on various tasks: P2 tried to *"support people interacting bimanually to find out if they are more accurate in drag and drop tasks, [or] faster"*, but also whether they would *"feel more personally involved in the interaction somehow"* (P2). This latter goal, of user experience or rich feedback, was the primary value for haptics:

> *"It's like having a touchscreen now on smartphones which nobody expects any other way anymore...sometimes pull out my old, uh, tom-tom navigation device in my car, and that one didn't have a touch screen back then (P3 laughs) so I tap on that one [expecting it to respond to touch input], and so it's the same thing with haptics, at some point it's just going to expect that you get some nice haptic feedback, but getting there is still a couple of years out"* (P3)

Of course, *"a couple years out"* has gone by as of the time of this writing; and indeed, haptic feedback is now normal and expected in many touchscreen products, although quality and range continue to be challenging.

As mentioned in Table 8.4.2, tailoring and customization are important for each implementation. This is also true for value: differentiable sensations are important to help distinguish overall user experience and provide branding. *look for alarms that were different; branding.effective, but different.* (P1). Companies and products need to have both a cohesive and differentiable feel. P2's company *"guideline book"*, which defined force profiles for buttons, was helpful to *"coin a trademark"* (P2).

> *"[We] provide differentiated tactile experiences to our customers, who are major mobile phone manufacturers. Since Android is pretty generic across the board, um, they like to have custom themes, which are sets of these 150 effects"* (P5).

With software libraries, themes are essential to the haptic design process. This desire for consistent output has a tension with customization and fine-tuning: *"it's also important to tune the experience depending on whatever kind of motor they decide to put in"* (P5). This is part of the the persuasive capability of touch: *improve comfort and differentiate based on branding* (P6).

CC5: *"A tough sell"* – Overcoming risk and cost. Despite its value, haptic technology is a risky, costly feature to add. Providing improved user experience requires *"high-definition haptics"*, not *"some rumble feedback that has been around a long time"* (P3). This often means *"going up in fidelity"* from a *"cheap, poor quality motor"* (P5). P5's company argues that *"the end-user is going to prefer this quality of experience"* with improved hardware, like a piezo actuator.

> *"[If we were to perform this project again,] I think we would spend a bit more time up front articulating the value, the specific value prop, of individual features"* (P5).

P5 notes the challenge of convincing non-end-users to buy or deploy their technology: *"[our company] has the unique challenge that our customers are not the people who use our products"* (P5). Since the main benefit is to the end-user's experience, it is challenging to connect to the bottom line, especially compared to other haptics components. According to P3, designers need to

> *"...get up to the decision-making level and more on the business side...[business roles] know nothing about technology, I mean, they don't care, but we are trying to demo parts to them, present business cases to them, and show them what they can do in order to gain market share, or increase their retail price when they add our technology"* (P3).

P3 further commented on lack of knowledge among dicison makers about haptics compared to other technologies.

> *"Let's assume we were to work on a completely different product like memory chips, so everybody understands what this is for, what it can do, and you probably have a memory chip that is faster or, whatever, smaller. Now for haptics, this approach is kind of difficult because the technology scouts themselves they kind of understand what this is for, but how it it's going to add value to their device, and how much they can increase the retail price, or if they can increase it at all, or gain market share, that's completely open"* (P3).

Newer technologies are hard to explain: *"[Gesture-based haptic feedback is] a much more complex task to design, and also to explain, to the OEM"* (P5). It can also make persuading a customer difficult. P3 finds that *"there's always discussions on the cost"*, and proposes *"alternative business models"* to no avail. Cost concerns are perfectly captured by P5:

> *"[The customer says,] 'we love [the piezo demo], it feels great, we're building this phone that has a 10 cent eccentric mass motor in it, can you make it feel the same?' The answer of course is no"* (P5).

P5 notes that *"cost pressures are pretty extreme"*. Mobile phones in the US cost *"$199 on contract, that's sort of a fixed price and you can add more features to the phone, but that just reduces the profit margin, right?"*, so *"the addition of haptic* *feedback technology...can be a tough sell"* (P5). Haptic technology is especially risky because of previously discussed challenges: it involves separate risk-adverse engineering divisions, and changes to the *"guts"* of a product. Designers need to set up complicated demos to persuade decision makers of the value of improved user experience: *if only compete on cost; then this is tough* (P1). Of course, *"it's hard to get through to the right level"*, like *"C-level kind of persons, so, talking to the CTO of Sony, those kinds of people"* (P3). The combination of high-risk, increased cost, and indirect connection to the bottom line make haptics a very tough sell indeed.

8.5 Part II: Validating the Findings in a Follow-Up Workshop

The second part of our study was conducted during a workshop on haptic experience design at World Haptics 2015, the largest academic haptics conference, held that year in Chicago, IL, USA (http://haptics2015.org).

8.5.1 Method

The workshop was organized by two of the authors to initiate a conversation between researchers and industry practitioners about HaXD status and needs, and elicit feedback on our findings in the interviews conducted in the first part of the study.

Participants

Over thirty people participated in the workshop brainstorm session and the panel discussion. Sixteen workshop participants responded to a questionnaire at the close of the workshop, which requested details about the respondents' roles, tools, and techniques.

Of 16 questionnaire respondents, 5 self-reported as working in industry, and the other 11 as members of academia (one reported also working at "other: research institute"). For roles, 4 reported as graduate students, 4 as developers, 2 as designers, 2 as a combination of designer and developer, 2 as researchers, 1 as a business person ("product integration/commercialisation"), and 1 did not report.

Procedure

Four leading haptic design experts – two from industry, one academic and one with a foot in both worlds – gave short *presentations* on topics concerning both engineer-

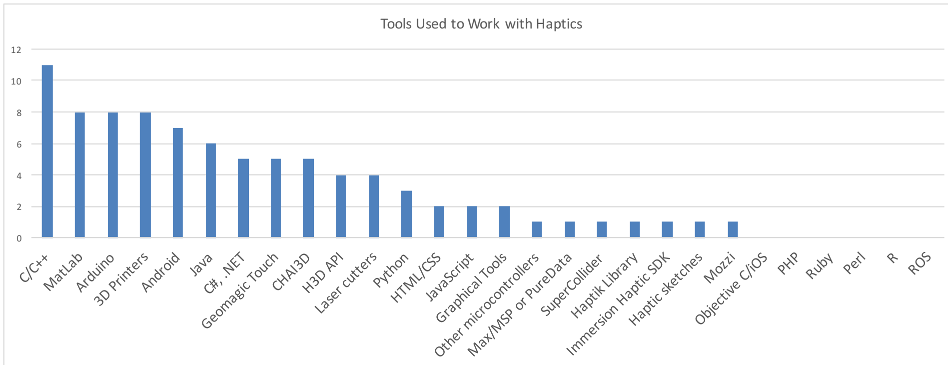


Figure 8.2: Responses for tools used in haptic design (N=16, *"check all that apply"*).

ing and UX. These presentations set the stage for a hands-on *brainstorming session*, culminating in an audience-participation *panel discussion* about challenges and ideas for HaXD.

Brainstorming occurred in 6 groups of approximately 6-7 members. Each group was asked to identify challenges faced by their members and then brainstorm solutions. Brainstormed ideas informed the panel discussion, which was led by the four haptic design experts but included all workshop participants. A questionnaire was distributed to all participants at the end of the workshop. The questionnaire was supplemented with researcher notes written during and after the workshop, and the participant's sheets used for brainstorming, which were collected afterwards.

8.5.2 Results

In the following, we report results from the questionnaire's quantitative and qualitative (open-ended) questions, along with findings from notes and brainstorming sheets.

Quantitative Data (survey): Tools, Background, Groupwork

Respondents reported a wide variety of hardware and software tools used to work with haptics (Figure 8.2). Most used were popular general or technical programming languages like C/C++, Matlab, Java, and hardware hacking tools like Arduino and 3D printers. Force-feedback APIs for consumer hardware (Geomagic Touch CHAI3D, H3D) were moderately used. Very few respondents reported using scripting or web tools, like Python, HTML/CSS, JavaScript, or more specialized tools. This combination suggests needs for performance, technical or scientific software libraries, and an ability to access and control prototyping hardware at a fine-grained level; in contrast to many other media design domains, web tools use is notably low. The latter is not particularly surprising for design that is, by itself, not primarily visual, and often comes with tight timing requirements.

Evaluation techniques were also varied (Figure 8.3); many respondents listed several. Most common were methods deployed in-lab or in-house (piloting, laboratory studies). Less common but still used were more externally valid evaluations (*in situ* studies and real-world deployment). Quantitative and qualitative methods were reported with equal frequency: 8 respondents reported using both, *i.e.*, a mixed-methods approach, and 4 respondents did not report either, but did report conducting in-lab, *in situ*, or real-world evaluation.

Group size reports suggested that hapticians work in groups with varying sizes (Figure 8.4). Few work in large groups; just one person (the designer / developer for a research institution) reported a group size of 21-50. No one reported a group of size 11-20, and most reported working with 3-5 others. Five participants reported varying group sizes (combinations of 1, 2, and 3-5 people). Because our question did not precisely define the meaning of a "group," we note the possibility of interpreting it with differing degrees of collaborative closeness, but assume that it was generally regarded as meaning the sharing community within one's organization.

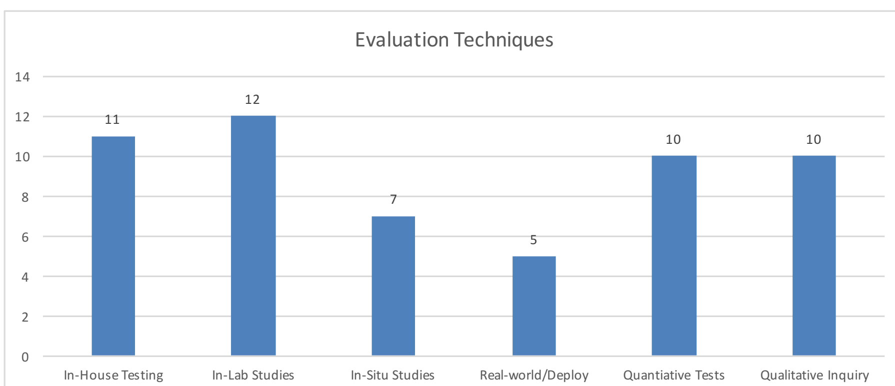


Figure 8.3: Responses for evaluation techniques used (N=16, *"check all that apply"*).

Qualitative Data (survey & brainstorming): Consistency, Quality, Value

Qualitative responses from the survey's first open-ended question, which asked for the largest challenges participants faced in haptic experience design, highlighted

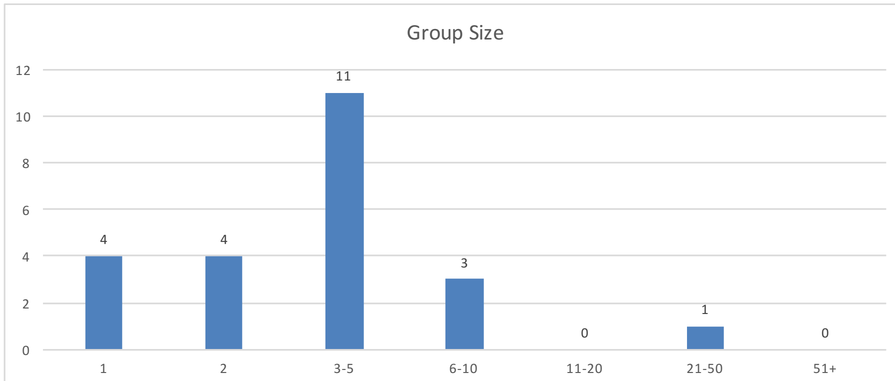


Figure 8.4: Reported group size for projects (N=16, *"check all that apply"*).

three themes. Each is a challenge for haptic design:

1. **Universal design**: *"universal experience"*, *"adopting wide spectrum of users"*, *"optimal and consistent delivery of haptic cues to large number of people"*, *"users variations in terms of subjective analysis"*, *"common haptic experiences, any person/any device"*, *"the spectrum of perception"*.

2. **Evaluating quality**: *"what is 'good enough'?"*, *"modeling haptic quality of experience"*, *"appropriate fidelity force feedback"*, *"optimal and consistent delivery of haptic cues..."*.

3. **Value**: *"getting people to realise the benefits of good haptics. Finding new ways to use hi-fidelity (wide bandwidth) haptic feedback to enhance UX"*, *"Bringing haptics to mainstream/consumer electronics"*, *"merging the technologies, make safety and pleasure experiences"*, *"convincing it's useful"*.

Other responses include emotion (*"transfer emotion through haptics"*) and language (*"haptic language; no simplicity in generating new sensations"*). In the second open-ended question, which asked what participants would like to see in a design tool to overcome these challenges, respondents suggested ways of handling variability or definability, such as automatic configuration:

> *"mapping"*, *"automatic evaluation of systems and actuators"*, *"quatification [sic] of haptic perception"*, *"autoconfiguration, calibration and prediction of results with the users"*, *"accessible to all, supported by standards"*, *"autoconfigure depends that perception [sic]"*, *"bigger testing pool"*.

Many of the challenges identified during brainstorming mirrored questionnaire results. Three groups (G1,2,6) focused on the *value* of haptics: what is the point or benefit in certain situations; as well as how to market that advantage to either a client or end-user. One group (G3) tackled the problem of *examples of good design*, including hardware and software architecture, suggesting a repository like GitHub or software engineering patterns. Three groups (G4-6) talked about *meaning* – and subjectivity therein, including the possibility of a shared or useful language.

The follow-up discussion developed these challenges further. Emergent themes were that haptics is not well marketed and that touch is taken for granted. The word "haptic" might be too jargonny or poorly understood; perhaps other terms like "tactile effect" or "physical effect" could be more useful. Curated examples and an on-line repository were offered as valuable goals.

8.6 Discussion

As a first step in further exploring the findings from our two-part study, we examine in more detail the critical activities practiced by haptic designers. This inventory confirms that HaXD is a field of design with familiar processes, but also one that is developing its own identity distinct from general UX design. We then identify major challenges encountered in HaXD that are unique to or exaggerated when the experiences being designed are haptic. We conclude with several concrete recommendations to support HaXD in the future and a vision for what this might look like.

We note that our interview results were generally applicable to professional haptic design in 2012. Based on recent interactions with designers in comparable roles (including those in the workshop), we believe our findings continue to hold at the time of writing (2016) with three caveats: (1) hardware has improved and become more diverse, (2) haptic feedback is more prevalent and more expected in the consumer culture, and (3) in a limited number of sub-disciplines, such as gaming environments and movie special effects editing, specialized tools have begun to appear to solve very specific designer problems where the cost/benefit equation merits their use. However, the design pressures shaping practices and tools themselves have changed little over that time, as confirmed by our findings from the more recent workshop (2015, Section 8.5).

8.6.1 Activities of Haptic Design

Based on our observations, we report the following activities that haptic designers conduct, all familiar to designers in other fields.

Develop and communicate vision. Designers must articulate the value that their designs can bring to both end-users and customers. They must communicate value to their team and others and, crucially, they must *persuade external stakeholders* that their product will contribute to the bottom line. To do this, they must *collect, run, and tune demos*, a critical part of the communicative toolkit for haptic designers.

Prepare for design. Designers need to *divine requirements* from customers, which customers often do not understand themselves. Designers also *gather examples*, both to provide inspiration and facilitate communication. Designers need ways to capture, modify, manage, find, use, and share examples and ideas, both ones they develop themselves and ones they seek out for inspiration.

Iteratively develop, communicate and evaluate multiple concepts. Our participants needed to iterate, often with their clients' and users' feedback to find the best designs. *Design thinking* and *user-centered design* are both important to apply to haptics, especially because requirements are difficult to communicate and understand. Additionally, designers must either *communicate with engineering*, articulating requirements to receive new physical prototypes, or *have engineering skills* to create demos and prototypes themselves. During iteration, hapticians must also *evaluate designs and collect feedback*, both with informal feedback from colleagues, and formal studies, typically run by a UX/research division. However, this practice is currently constrained both by industrial concerns (confidentiality, cost, end-user access) and the hard-to-share nature of haptic technology itself.

Interface with research. Hapticians need to *hand off* prototypes or stimuli to their UX or research division, and communicate study goals. They must also *monitor the academic research* in this rapidly changing field, interpreting data emanating from multiple sources: marketing research, psychophysics studies on hardware and stimuli, and interaction design of applications. Alternatively, as with engineering, they might *plan, run, and analyze studies* directly.

Manage IP. Designers must be sensitive about intellectual property, both that of their company's technology, and of the many companies and divisions they interact with. This can involve deliberately *exploiting or avoiding particular design approaches*; and has heavy implications for *confidentiality and privacy* of their overall process.

8.6.2 Challenges for Haptic Experience Design

From our interview and the subsequent workshop findings, we identify several challenges facing designers that are unique to HaXD or are exacerbated when working with haptics compared to non-haptic UX design.

Context is largely unknowable

Haptic experiences are multimodal and holistic, interacting closely with physical hardware, grip, and orientation. When our participants knew the haptic experience's physical context, like the dashboard of a car, they were able to use tricks to improve designs and circumvent constraints. However, when context is unknown, *e.g.*, due to confidentiality, diverse environments, applications and means of handling an interface, hapticians can be quite hampered in their attempts to create consistent experiences.

Applications and individuals vary a lot

There is no "one size fits all" in haptic experience design. Each customer's design challenge has new properties. Hapticians must continually adapt their practice to changing conditions [198], and cannot simply design once and deploy. Companies use haptic sensations to brand their products, and individuals might want to customize effects for their preferences: users perceive, understand, and respond affectively in different ways to a haptic experience.

Demos are complex, costly, and crucial

Essential in eliciting requirements, communicating vision, and persuading customers, demos are hard to manage. With many moving parts and ways to fail, demos often require a dedicated assistant; latency is a special challenge for early prototypes, and can defeat carefully synchronized multimodal effects. Because HaXD takes place over global distances, and across organizations and disciplinary boundaries, it is often difficult to have a handler onsite, send proprietary hardware, or divulge enough detail for clients to run them on their own.

Iteration is painful

Every change to a haptic experience results in a change to the "guts", including reinforcing modalities and physical setup; technical constraints are tight and unyielding. Hence, even early 'sketching' iterations to understand requirements can be slow and difficult, limiting playful exploration of a design space and disrupting communication with customers and users.

Barriers to interdisciplinary collaboration are significant

Hapticians either need to fill many roles or work in groups that include hardware, software, design, business, and psychology. Furthermore, haptic design teams must interact with many external stakeholders stratified across different international companies, encouraging remote, asynchronous collaboration with physical, synchronous designs.

Cost/benefit ratio is not obvious or easily quantifiable

The benefits of haptic technology are often intangible: better user experience, usability, branding, and sense of quality. Product manufacturers (phones, cars) must be convinced of the contribution to the bottom line, and are all too aware that improving haptics comes with increased cost and risk. Haptic design teams reaching out to customers through risk-adverse engineering avenues face additional pushback.

Evaluation methods are limited and often not pactical

Quality of experience, usability, and branding are difficult to study with physical systems. Although many of our participants mentioned evaluation methods as important, time and cost constraints limited it in practice; acceptance testing seemed to be the primary tool. Hapticians use both qualitative and quantitative methods, but *in-situ* evaluations are difficult to come by, suggesting that haptic designers primarily conduct evaluations in-lab and do practical deployments.

8.6.3 Recommendations for Haptic Experience Design

From these challenges, we identified three main directions for development which could lead to better haptic design in the future.

Develop adaptable haptic interfaces

Many of the challenges facing haptic designers are a result of uncertainty or variability in physical context. One solution is to let physical haptic interfaces adapt their context, either automatically or with help from a designer, customer, or end-user.

One automatic approach to mitigate variable physical context is to employ *closed loop control*: adapt actuator output to desired levels with sensors. For example, a microphone could sense the external vibrations of a VT actuator, whose output can then be modified to overcome the effect of external factors like material, orientation, and grip and achieve a specified frequency, amplitude and responsiveness. This might be deployed in products during use, or once during manufacturing as quality assurance to adapt for different product materials.

Another approach is to let the customer or user adapt the experience through *customization*; this can handle both physical context and individual differences in perception and preference. This might be a simple volume control, or a powerful menu of settings. Customizable infrastructures that support fine-tuning can also help speed iteration once demos or even fully-fledged applications are set up, letting designers and customers try variations of a haptic experience easily.

Finally, *efficient calibration* of demos, using either sensors or a person's input, could improve collaboration with easier demo or product setup. Devices that are self-assembled or operable by non-experts require an easy way to troubleshoot and ensure correct rendering. This could engage the DIY community to explore haptic technology, and improve efficacy of sending evaluation kits to potential customers.

Exploit virtualization

The unique problems of haptic design stem from the combination of physicality and the software engineering necessary to integrate the hardware into a solution. Some of these challenges may be offloaded through virtualization: certain types of iterations or tests can be done more efficiently with software simulations or crowdsourced evaluation – once this capability exists.

Proxies are one way to virtualize complex physical setups, *e.g.*, using low-fidelity feedback like phone vibrations when high-fidelity feedback is unavailable [197]. Low-fidelity previsualization of haptic sensations (or "pre-feels") [196] can improve iteration speed, by allowing the designer to experience an approximation of an iteration before committing resources to building it, and/or to compare with a reference starting point. Visual or audio proxies can easily exploit existing infrastructure.

Software simulations of hardware can explore how different electronic or mechanical components could be rearranged to preserve or enhance dynamics, reducing physical prototyping. Even more advanced might be the use of simulations to develop "perceptually transparent" sensations [**?**], allowing actuators or other components to be swapped in and out if upgrades or cheaper models are available, while software components are automatically updated to achieve a consistent end result. This virtualization technique dovetails nicely with closed-loop adaptable interfaces by establishing models and correcting for errors.

Software has enabled immediate, efficent deployment of visual and audio stimuli through the Internet. Analogous *infrastructure* could help haptic technology catch up to other modalities more quickly, *e.g.*, developing modular systems, data structures and protocols, and large on-line repositories of examples. *Broadcast* haptics remains an important and unrealized goal, which can help both with potential customers and end-user experiences [153].

Establish richer conceptual infrastructure

Several measures can help to address communication and cultural barriers to haptic design.

Outreach and education might be able to improve perceived value of haptics and facilitate interdisciplinary communication. Public haptic portfolios, accessible haptics education [114] like online tutorials, support for DIY and maker cultures, and events like haptic hackathons [**?**] will help to establish haptics as a known term, spread the word about its value, but more importantly help more people join the conversation in which we will articulate the value in touch-based technology. It will help provide different stakeholders with common reference points, language, and understanding, both lowering the bar to conduct haptic design as a team member, and by providing a voice to external stakeholders.

A *haptic design language* is needed for multidisciplinary team member and client communication (C6). Much like graphic design, where non-experts might be aware of some concepts (symmetry, contrast, hot/cool colours) while experts know much more (colour combinations, concept of weighting in a visual design), a shared, objective and teachable language will help teams communicate across divisions and with clients, users, and customers. It remains to be seen whether this will be a formal lexicon of terms, or ideas that emerge organically; either way, we suggest paying careful attention to the language used when doing haptic design, and to share the language alongside the sensations and their components.

Haptic designers have limited access to *evaluation techniques* that are taken for granted in other modalities, especially *in situ* tools. Promising ways of mitigating this handicap is application of remote analytics to haptic design, *e.g.*, logging, machine learning, or qualitative contextual inquiry. This may require development of new batteries of haptics-suitable tests, especially ones which target its less objective benefits (*e.g.*, quality and branding), which might in turn help to study perceived value and risk.

8.6.4 Future of Haptic Design

Hapticians follow an observable, defined process. They collect requirements, develop multiple concepts, and iterate until they arrive at a final experience, which is then evaluated with varying amounts of rigour. We saw evidence of libraries, examples, and our participants' own craft and experience; we also saw a diverse, international, collaborative ecosystem. Some deliberately applied user-centered design techniques.

However, we also see that haptics *"might be 30 years behind graphics"* (P3), or at least *"really new"*, *i.e.*, in an early stage of development. We believe that HaXD can draw from both newer fields like experience and interaction design, as well as more established ones like graphic design. How might it look?

Haptic designers might work in teams, interacting with other relevant units. From our research, it is likely that haptic designers will need to communicate with everyone from mechanical engineers and software developers to people conducting business and user research. As with graphic design schools, there may be formal education available for haptic designers. However, as haptic technology needs to be tailored to each specific problem, these will likely be generalized professional programs that train diverse skills, or will focus on certain sub-categories of haptic technologies, *e.g.*, tactile artists or animators [196], friction designers, or 3DOF force-feedback developers. As hardware becomes more affordable, we also expect the recent Maker movement will encourage hobbyists and artists to explore haptic technology and push its limits.

As with other emerging media, such as the web browser wars of the 90s, standardization of HTML/CSS, and Blu-Ray versus HD DVD, we expect diverse file formats and infrastructures to emerge and then coalesce. Given the diversity of haptic technologies and experiences, we expect these to be centered around *paradigms*, mental models of how to work with a haptic experience. For example, haptic icons [140] are one paradigm: display-only, temporal and meaningful entities rendered on a single body location. These might be designed, distributed, and experienced similarly to audio files. Tactile animations [196] are another: generalized spatio-temporal entities that can be rendered continuously on different grids. Multi-DOF force-feedback displays are often programmed with a third paradigm: a virtual environment and a single manipulator; this is most analogous to 3D virtual worlds. Paradigms can be applied to multiple devices in a class (*e.g.*, tactile animations on grid displays), or multiple paradigms might apply to a single device (a Haptuator [233]) can display a haptic icon (temporal only), or it can produce a directional force [**?**] (spatio-temporal).

We expect design dimensions to be further developed, and eventually encapsulated into best practices, just as alignment, contrast, and weighting are used for graphic design. Other design languages, like musical notation, will facillitate recording and communication amongst experts. Meanwhile, more developed aesthetic theories, like musical or colour theory, will help guide people to effective, pleasing, differentiable haptic designs. Intellectual property law will need to be adapted – much like a logo can be trademarked, how might a certain button click? Whether a haptic icon set should be protected, and how to set an appropriate level for burden of proof, remain open questions. We hope that these questions and more will be answered during this exciting time for one of our most essential senses.

8.7 Conclusion

In this paper, we provide a first picture of how haptic experience design (HaXD) is being practiced in industry. We report findings from interviews with six haptic experience designers, finding observations about designer process and themes about the holistic nature of haptic experiences and the collaborative ecosystem and cultural context of our participants. We supplement this with broad follow-up data from a recent workshop at a major haptics conference.

We identified the various activities haptic designers practice, similar to other fields of designer. We also note specific challenges facing designers who work with haptics, and recommend both high-level priorities and low-level tactics for conquering those challenges. This contribution is a first step in understanding HaXD outside of the research lab; we look forward to when physical, interactive technology can be designed with creativity, passion, and panache.

8.8 Acknowledgements

We deeply thank our designer participants for their time and dedication, Gordon Minaker for helping with survey distribution and workshop administration, and Hasti Seifi for helping to conduct the workshop brainstorming. More broadly, we thank generations of our own lab's designers and our collaborators for the foundations they laid under these findings and ideas. This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), the GRAND Network of Centres of Excellence, and the University of British Columbia's 4YF fellowship program. The user study was conducted under UBC Ethics certificates #B01-0470 and #H13-01620.

Chapter 9

Conclusion

In this dissertation, we explored haptic experience design (HaXD), looking at its process to inform how to design, build, and evaluate HaXD support tools. In this chapter, we summarize our research contributions from the preceding chapters and provide final thoughts about the HaXD process and support tools. In particular, we discuss process, including challenges and strategies haptic designers can use to overcome those challenges, and findings with specific implications for designing and developing software tools to support HaXD. We conclude with directions future work and final remarks on supporting HaXD.

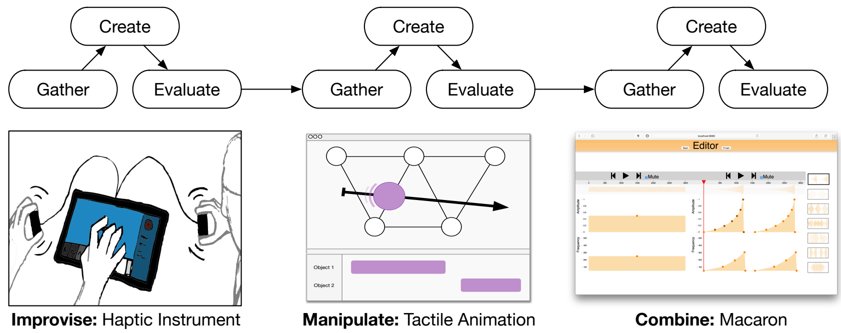
9.1 Summary of Research Findings

We summarize our more specific findings organized by approach: depth, breadth, and grounded capstone.

9.1.1 Depth: Vibrotactile Design Tool Case Studies

Through Chapters 3-5 we designed, built, and studied a trio of HaXD support tools. Each case study investigated a different set of design concepts with varying user populations, VT device, and design challenges (Figure 9.1). Each followed three design steps: *gather*, finding requirements and previous design elements; *create*, where we design and build the tool; and *evaluate*, where we test the tool with its target population and consolidate lessons learned. We include HapTurk (Chapter 6), a VT *sharing* technique, in this discussion. We began with an initial hypothesis: that real-time feedback and collaboration could improve the haptic design process. Through our tools, this hypothesis was confirmed, but more importantly, elaborated and refined.

Initial Exploration: The Haptic Instrument (Chapter 3) – In Study 1, the



Who	Haptic Designer-Proxies *(researchers with some haptics experience)*	Animators *(visual designers who would be haptic designers)*	End-users *(people with and without basic media editor experience)*
What	Dual Haptuators	Spatial Vibrotactile Pad	Wrist-worn C2 Actuator
How	Real-time, collaborative parameter control	Direct space/time manipulation of animation object	Example-based interaction techniques, logged user actions
Discovery	Good for ad-hoc exploration, collaboration; bad for refinement	Real-time manipulation valuable; animators drew from examples	Open examples directly facilitate design and indirectly scaffold learning for VT design
Platform	Mobile (Android)	Desktop (Python)	Web (JavaScript)

Figure 9.1: Vibrotactile design case studies. Each studied an aspect of vibrotactile design with a varied set of users, devices, platforms, and foci.

Haptic Instrument, we focus on real-time, rapid design of VT sensations with a first look into themes of real-time design and collaboration. Our implemented haptic instrument, mHIVE, showed us that rapid exploration was possible with real-time feedback, demonstrated value in informal, collocated collaboration, and gave evidence that showing rather telling about haptic sensations could circumvent an impoverished language (*e.g.*, what do you think of *that*). mHIVE also showed us that there are distinct roles to be played by different tools: mHIVE was successful for early *sketching*, but did not enable *refinign* and had a learning curve.

Direct Manipulation Pipeline: Tactile Animation (Chapter 4) – Our second tool, Mango, followed up on these themes with a focus on implementation. We established a rendering pipeline for both design and playback. A direct-manipulation metaphor let participants both *sketch* in real-time, and *refine* their designs. In addition, an animation paradigm enabled our visual animators to transfer their skills to tactile animation. Our evaluation study found evidence for reuse: repetition played a large role in tactile animations, and participants again drew inspiration from their experience or external examples (*e.g.*, a YouTube video of a heartbeat) - and noted the engagement of multimodal design, *e.g.*, designing for an audio clip. These both informed Macaron.

Example Use and Analytics: Macaron (Chapter 5) – With our third tool, Macaron, we were able to easily implement the system (drawing from Mango's architecture), allowing us to study our participants' process more closely. We found a number of concrete recommendations for HaXD tools, and confirmed the value of *browsing* examples: we found different strategies for using examples in the initial design process, and open or visualized examples helped designers learn how to conduct VT design. Macaron also helped us find our more nuanced understanding of our initial hypotheses. Real-time feedback is useful as a preview, to get the right frequency, amplitude, or timing. However, participants would also step back to feel the entire design in its entirety, something that can be done

Feedback at Scale: HapTurk (Chapter 6) – While not an iterative design tool study, HapTurk is a focused investigation of a VT design technique for collecting large-scale feedback on VT icons. HapTurk (Chapter 6) allowed us to study collaboration more thoroughly, as we focused on other aspects of design with Mango and Macaron. We found that VT icons could be *shared* over MTurk to elicit large-scale feedback from remote users. We also found affective qualities of VT icons could be communicated through proxies, suggesting we can express haptic ideas in different modalities.

9.1.2 Breadth: Focused Haptic Design Projects

Together, mHIVE, Mango, and Macaron have informed us on both important features and roles for design tools, given us insight into implementation and evaluation, and helped to study HaXD as a process. HapTurk extended this investigation to collaboration. To broaden our understanding of haptic design, we undertook several more focused haptic design projects to look at different activities, application areas, and haptic modalities. In **Chapter 7**, we describe several smaller projects that gave opportunities for practicing haptic design and exploring other types of haptic feedback:

FeelCraft (Section 7.1) is a plug-in architecture that augments media with customizable spatial VT effects. FeelCraft explored existing infrastructure for haptic media, and to design VT effects for a popular video game, MineCraft.

Feel Messenger (Section 7.2) is a chat program augmented with expressive, customizable VT effects using commodity hardware and APIs.

RoughSketch (Section 7.3) is a painting application for the TPad Phone, a variable-friction mobile device, for the World Haptics 2015 Student Innovation Challenge. Variable friction is a significant contrast to VT sensations as it is intrinsically connected to input: no sensation can be felt without active movement by the user.

HandsOn (Section 7.4) is a conceptual model for creative education software using low-cost, DIY haptic hardware, giving us an understanding of how to work with 1-degree of freedom force feedback and an educational context.

CuddleBit (Section 7.5) is a project inspired by the Haptic Creature [38, 236, 237] and CuddleBot project [2]. We use small, breathing robots to explore the display of emotion, and extend our findings from VT design tools into new tools for this modality: Voodle and MacaronBit.

9.1.3 Ground: Data from Haptic Experience Designers

In **Chapter 8**, we complement our design-based inquiry with data from haptic experience designers, or hapticians. Haptic designers are difficult to access as they are both rare and often work in proprietary contexts, with limited ability to speak about their process. To include hapticians' perspectives without severely hindering our design tool development, we conducted a grounded theory analysis of interviews with six haptic designers. We characterized cross-cutting themes at three levels of scope: 1) the holistic nature of haptic experiences; 2) the collaborative ecosystem of haptic designers and related stakeholders; and 3) the larger cultural context of haptics in the public consciousness. We augmented these interview findings with those from a workshop organized at a major international haptics conference, World Haptics 2015, which let us connect with designers more widely.

Our findings showed that haptic designers follow a general user-experience design process, but with added challenges because they work with haptics. We articulated these challenges, concrete recommendations on how to make progress on them, and finally gave a vision of haptic experience design as it might be practiced in the near future. These results overlap with our findings from our design-based inquiry: some conclusions are confirmed, with similar themes emerging both from expert hapticians and our design studies; others emerged from only one source.

9.2 HaXD Process: Requirements for Tools

In this section, we examine our findings about the HaXD process. In Chapter 8, we saw that haptic designers follow a familiar design process. However, we also saw that there are unique challenges that differentiate HaXD from other modalities of design, which are confirmed by our work into HaXD support tools and our focused design projects.

We begin by discussing four design activities that occur generally in design, but need to be explicitly supported for HaXD: *browse*, *sketch*, *refine*, *share*. Then, we comment on some approaches for handling the diverse devices and modalities employed by haptic technology. After, we discuss some techniques to imbue haptic experiences with meaning and realism. Finally, we talk about the importance of customization and how to support it.

9.2.1 Contextual Activities of Design: Sketch, Refine, Browse, Share

In our first exploration of HaXD support tools, the Haptic Instrument (Chapter 3), we found evidence that mHIVE was able to support exploration and sketches of haptic ideas, but not refinement into final designs. mHIVE was also able to support collaboration in certain ways. In our followup studies, we explored these activities that draw upon a designer's context, eventually arriving at four that we found are valuable when thinking about tool design: browsing examples, sketching new ideas, refining existing ideas, and sharing ideas with others. These activities can occur at any point in the design process, and we do not propose them as an exhaustive list; for example, "framing" [198, 227] could be claimed as an activity for design which may overlap with activities like Browsing and Sketching. We focus on their utility in motivating features and specific tools to aid haptic experience designers.

Sketch

Sketching allows people to form abstracted, partial views of a problem or design, iterate very rapidly and explore concepts. The generalized notion of sketching can support other activities: sketches are a notation that can be shared, and provide a vehicle for annotating and iterating on designs. Here, we use *sketch* in contrast with *refine*, to refer to embodied exploration, concept generation, and initial design.

Early in our exploration, we found that our Haptic Instrument, mHIVE, excelled for exploring a design space but faltered for refining sensations. A real-time direct manipulation model in Tactile Animation facilitated both. In Macaron, we observed different levels of exploration and refinement, discovering a pattern of focused adjustment and repeated reflection [198], where the designers stepped back to feel their design in its entirety, and zoomed in to adjust parts in detail. We find this is a repeated pattern, where participants iteratively zoom in and out to different levels. The mixture of focus also depended on the stage of design - early on, exploration and execution take a more prominent role, while later refinement has smaller adjustments and more reflection. We have found this distinction useful when developing suites of tools, most explicitly when supporting design of CuddleBit behaviours: Voodle is a free-form *sketching* tool, which can hand off to MacaronBit for *refinement*. Future work remains to be done to establish whether there are discrete levels of focus, or if they lie along a continuum. In general, we've found two main features important for supporting Sketching: *abstraction and ambiguity*, and *rapid iteration*.

Abstractability and ambiguity – Haptics suffers from a dearth of notation. Sketching of physical devices or interfaces is well supported, with paper and pencil and innumerable software assists. Sketching *motion*, and in particular showing what is or might be felt in, say, a vibrotactile experience, is trickier. While we can sketch a visual interface and look at it, it is much harder to sketch a haptic sensation and imagine it without feeling it.

Rapid Iteration – Design requires fluid, playful interaction with potential designs and the associated problem space. Haptic design sometimes requires fuzzy goals like feeling "just right"; these types of problems require designers and users to feel variations until they get the experience right. When iteration is slow, it is painful and distracting.

Refine

Design requires iteration to *refine* an initial set of ideas into a single well-developed one through concept generation followed by iterative revision, problem-solving and evaluation, until only small tweaks are necessary. This long view of the design process is necessary to see designs through to the end; further, tweaking final designs is a valuable way to accommodate individual differences, and was adopted by our haptic designers in Chapter 8.

Incorporating haptic technology into a design is an extremely vertical process, dependent on specifics of hardware, firmware, software, application, and multimodal context. With the complexity of these many components, there can be a significant initial cost to setup a first haptic experience; then, adding this complexity to the time needed to program, recompile, or download to a microcontroller means iteration cycles have the potential to be slow and painful. There are two stand-out features to support refinement: mature, polished tools, and adaptable interfaces.

Mature design tools – Implemented design tools a glut of features, *e.g.*, picture the sheer number of commands, shortcuts, and organizational scaffolding supplied by image editing software like Photoshop. This power and precision can improve refinement, especially when integrated into haptic media pipelines to fluidly move initial ideas to final experiences. We describe the road to mature tools in Section 9.3.2.

Adaptable interfaces – Calibration, customization, and sensing were identified in Chapter 8 as one major approach for handling the varied end-user context. These techniques will be essential to ensure consistency and quality of refined sensations, or to let designers and users alike fine-tune their experiences. We describe these in more detail in Section 9.2.4.

Browse

"Browse" can have specific meanings for interacting with data [159]. Here, we use *browse* to refer more generally to the act of looking at examples, *e.g.*, corkboards of previous designs [27]; drawing from previous personal and professional experiences, *e.g.*, one's repertoire [198]; or real-life sources to inform a design. We found this emerged in different ways: in Chapter 3, participants used their personal experiences to interpret sensation meaning (*i.e.*, schemas, discussed more later); in Chapter 4, one animator brought up a YouTube video of a heartbeat to ground his animation; in Chapter 8 several participants described collecting examples or using guide books.

We highlight this activity because haptic designers encounter modality-specific barriers when gathering, managing, and searching for examples. Explicitly supporting browsing can make a difference: in Chapter 5, we found visible, incorporable examples both eased design and helped scaffold learning for non-experts. We believe scaffolding to be extremely important, as there are few haptic designers practicing today. Browsing is also tightly associated with the ability to Share designs and collaborate; when one designer shares their designs or ideas, others are able to browse it. We highlight three main challenges to supporting browsing in HaXD: representation, organization, and access.

Representations of single sensations: – How do we store, view, and organize haptic experiences? Haptic technologies are often inherently interactive, part of a multimodal experience with visual and audio feedback, and can take a variety of physical forms depending on the output (and input) device. This last point is particularly bothersome should the user not have access to the original device type – imagine trying to browse force-feedback sensations on your phone!

Collection classification and organization: – Haptic language and cultures of meaning are still in active development. Without a commonly shared lexicon, organization dimensions, or even adjectives, it is difficult to curate collections. Compare this to sound: most musical terms have a long tradition with a clearly defined lexicon (*e.g.*, crescendo, staccato); non-musical sound effects generally "sound like" something, and are often literal. With vision, one does not have to be a graphic designer or artist to instinctively understand "warm" and "cool" colours; the color wheel is introduced to us in grade school.

Overviews, search, and skimming: – Collections of examples, especially visual or physical collections, are often displayed spatially for ambient reference or to enable quick scanning. When you can't feel multiple things, it can be hard to get the big picture or swiftly peruse a collection. Both designer and end-users have needs for finding similar/different vibrations in a collection, requiring a low barrier-to-entry on any overview technique.

Share

Sharing designs is valuable at different stages of the design process [125], whether for informal feedback from friends and colleagues, formal evaluation when refining designs, or distributing to the target audience for use and community for re-use [206]. As haptic experiences must be felt, this process works best when collocated with only a few collaborators, whether by having collaborators work in the same lab, or by showing final experience in physical demos. During ideation, ideas can be generated when collaborating remotely, but physical devices need to be shipped back and forth and it is difficult to troubleshoot and confirm that configuration and physical setup are the exact same. Feedback also typically needs to be collocated, using in-lab studies or feedback, or shipping devices between collaborators. Furthermore, visual and audio design support very easy capture of ideas to share later, through smartphone cameras and microphones, that could later be browsed. We suggest two directions for future work: easy capture of design ideas to share for later, and remote sharing through proxies.

Capturing ideas – Inspiration can strike at any time. In order to *browse* ideas later, or to snap a haptic picture and *share*, we need advanced haptic cameras [142]. Repositories are only useful when they are populated; easy capturing methods can encourage crowds to upload their own content for later use, perhaps leading to stock haptic experiences (like stock images) and a viable venue for freelance haptic designers.

Remotely or asycnhronously share haptics – Touch is a proximal sense, and difficult to share asycnhronously or over large distances. Techniques like proxy sensations (Chapter 6), easy calibration (Section 9.2.4), and fabricated haptics using, *e.g.*, 3D printers [222] can all help *share* these physical sensations around the world.

9.2.2 Generalizing Devices and Sensations

One major challenge facing haptic designers is the variety of haptic devices available. Each device has different physical properties, and may use different actuation principles or sensory modalities to communicate with the user. This might be analogous to how modern web sites employ responsive design, adapting to different screen sizes, albeit more extreme. A screen, at the end of the day, is that plane with a given physical size and pixel size; with haptics, contextual problems like grip can influence feedback, and haptic feedback can vary from single VT actuators to VT grids, programmable friction, skin stretch, or force-feedback devices. We suggest three ways of managing this complexity: Grouping devices and interactions into *paradigms*, considering representational *translations*, and using affordances and closed-loop sensing to create *consistency*. A fourth related strategy, enabling customization, is so important we discuss it later.

Paradigms

However, we believe this problem is more diverse than screen size and resolution. We suggest this is a single *paradigm* for graphic designers, analogous to how haptic designers might work with several different types of single vibrotactile actuators. Haptic design becomes more diverse when designers need to consider arrays of VT actuators, programmable friction, skin stretch, or force-feedback, all of which are as analogous as screens are to, e.g., speakers.

First captured in Chapter 8, we believe that paradigms is a key concept to designing HaXD tools. We define a "paradigm" as an abstracted model of how to work (?) with a haptic device. Much like how different programming language paradigms, such as functional or object-oriented languages, can accomplish similar results but enable programmers to think in different ways more appropriate to their problem, we believe that different haptic or multimodal paradigms will enable different problem solving techniques for HaXD. In our in-depth studies we present three: an instrument paradigm, an animation paradigm, and a track-based editing paradigm.

It is critical to note that there is a many-to-many mapping between paradigms and haptic devices. Tactile animation can be used for multiple spatial grids and track based editors are generalized enough to handle multiple display types. Meanwhile, a multi-VT grid could be controlled by any of these three paradigms. As haptic displays become more diverse, we expect paradigms to play a larger role for organizing design perspectives, and multi-paradigm tools to become successful, just as multi-paradigm programming languages like Python and JavaScript afford flexibility, power, and accessibility - providing increasingly low barriers, wide walls, and high ceilings.

⌈**OS** *Paradigms can be split into temporal, spatial, and interactive (reactive) elements.*⌋

Representation and Native Platform

One striking problem with haptic technology is its sheer diversity. There are a wide diversity of devices, many of which support different paradigms, and all of which can be different based on their physical configuration. This poses a problem of access - if a designer creates a force-feedback sensation, how do you render that on a mobile device with a simple vibrotactile actuator? Can this translation be done?

Compare this to graphic design. At the end of the day, most graphic designs will be on a 2D plane, whether a screen or not. There are different sizes, resolutions, and colour maps - for example, print media might be designed differently than web - but similar tools and principles apply. In haptics, we might apply these different sizes and resolutions to a class of device - for example, a VT actuator can be an expensive C2 tactor or a low-cost voice coil. However, a single actuator is quite different from a 2D grid of actuators (like in Tactile Animation), and dramatically different from variable friction feedback (like with Rough Sketch) or 1-DoF force feedback (like with HandsOn). The question of resolution and platform within a single class of devices is analogous to the challenges faced by graphic designers and sound designers, but the diversity of classes of devices is even more varied.

Consistency through sensors and context control

A third way to adapt to the variety of contexts a haptic device might be used in is to either impose a known context, or sense and adapt an uncontrollable context.

Our industry haptic designers would talk about working with automotive companies, and how the material in the dashboard could affect the final haptic sensation. By controlling this material and working in a known environment, haptic designers might be able to keep their designs more consistent: When actuating a touch-screen in a car, a designer could know the materials. Similarly, wearable devices like the wristbands (Pebble, Apple Watch) know where the device will be worn, and can use that knowledge to their advantage; they can also use the materials of their wrist-straps to ensure a reliable tightness or pressure on the skin. Force-feedback devices like haptic knobs might change their handles, using physical affordances (Gibson? Norman?) to suggest a grip to the user.

Alternatively, closed-loop sensing might be able to standardize sensations. Blum et al. [13] showed that accelerometers can provide insight into perceived loudness of VT stimuli. Techniques like activity classification (*e.g.*[195]), or even vibration sensing of a VT effect could help reduce noise from physical changes and material properties. This could be done *in-situ*, or during manufacturing for different device materials as a quality assurance step.

9.2.3 Framing and Meaning

Because of the novelty of haptic design, we found that conceptual framing is important for understanding the intent of haptic sensations. ⌈**OS** *TODO: [24]*⌋ We have suggest three ways of designing meaning into haptic experiences: schemas and metaphors, design languages, and reinforcement through narrative context and other modalities.

Schemas and Metaphors

Schemas are existing conceptual models used as transitionary objects to understanding new concepts [170]. We found this procedure occurred not only in educational contexts, but also in design. In our early Haptic Instrument exploration, we found our participants' prior experience was a lens through which they interpreted haptic sensations. For example, one cat-owning participant interpreted likened sensations to cat purring, while another drew more on their knowledge of engines and cars. Heartbeats and rain [108] are effects that were easily understand by general participants, and verified using perceptual studies. Schemas are useful both for framing user interpretation of haptic experiences, and for designers themselves. As covered in Chapter 2, many previous systems have their roots in other, non-haptic concepts, *e.g.*Touch TV []. This enables transfer effects; as we showed with Tactile Animation (Chapter 4), visual animators were able to easily create tactile designs.

Design Languages

Another way of framing a haptic experience is to use a *design language* like Google's Material Design (https://material.google.com). A design language is a defined set of aesthetic and interactive rules to ensure a consistent look-and-feel. We also believe that Gestalt-like principles will play a role in defining the possible design languages for haptics, much like musical concepts of thematic development, restatement, elaboration, and expectation, or graphical design concepts like contrast, repetition, alignment, and proximity are the tool with which languages can be formed. Previous efforts have identified frequency, amplitude, rhythm, affect, and spatial location as the main design elements for vibrotactile design. We have started to identify similar, middle level concepts - alignment and repetition emerged in the Macaron study.

Narrative Context and Multimodal Reinforcement

The intent of a haptic sensation was not always interpretable from the sensation itself. A great deal of the interpretation of an effect depended on the narrative context. Previous work used linguistic descriptions, such as "light rain" [108]. In our FeelCraft and Feel Messenger studies, we found that linking adaptable sensations to in-game stimuli was effective. With Feel Messenger, we at first attempted feel effects with abstract icons, such as "motor" for a rumble effect. In early piloting, these were not very helpful for understanding. Adding in vibrant cartoon emoji icons, and using a cat's face (for a purr) rather than a motor image were much more effective in sending haptic icons.

It is also worth mentioning the role of realism in a haptic experience. There are two types of realism, photo-realism (veridicality) and real-seeming (verisimilitude) [150]. For example, cartoons and animation are clearly not depictions of reality (they do not display veridicality), but they can appear life-like (they display verisimilitude). In this dissertation, we focused on verisimillitude rather than veridicality; the latter is often already accommodated by realistic rendering techniques. We believe both require a design process, but more photo-realistic approaches have different challenges than cartoony ones, and may require more of an engineering perspective than a design perspective. That said, sometimes combining the two is helpful; much like how in computer graphics, physical simulation and realistic rendering are used as a baseline, but artists still may want artistic control over certain scenes or effects.

9.2.4 Adaptable Interfaces

We've shown how individual differences are a prominent feature of haptic perception and psychology. Furthermore, variability in and poor designer control over context – user attention and device form factor and manner of connection, as well as use environment – mean that haptic sensations often need to be tuned to both each person and each use case. In Chapter 8, we suggested that adaptable interfaces could help manage changing context and encourage consistency and appropriate feedback.

The most likely solution is to make it easy for end-users in populations that might benefit from a haptic modality to *customize* aspects of haptic design elements, whether by choosing pre-formed settings and "skins", adapting defaults, or wholly designing their own. Possible approaches range from volume-like slider controls, options to select sensations from curated collections, or, at the more complex end, perceptually-confirmed filters like those found in Instagram or PhotoShop [188, 200**?**].

9.3 HaXD Tools: Designing and Implementing

In this section, we comment on the specific features and forms HaXD support tools might involve. Because of the diverse activities described in the previous section, we believe there is no silver bullet: haptic design tools are likely to form a suite or ecosystem. Here, we discuss ways to enable creativity through mature tools with "a low barrier, wide walls, and a high ceiling." We talk about important collaborative techniques and how to deploy implemented tools. We discuss implications for developers and software engineering teams. Finally, we conclude with some comments on our evaluation methodology and how future HaXD tools might be evaluated.

9.3.1 Communities and On-line Deployment

Unfortunately, haptic technology faces obstacles to *browsing* and *sharing*, especially when under development; this has typically confined distribution and exposure to lab prototypes available only to in-person visitors. Demos are difficult to setup and there is little infrastructure in place for distribution. While synchronous, collocated collaboration is effective with demos, remote or asynchronous collaboration is rife with trouble. Consistency must be maintained, and setup can be painful for those not trained in setting up devices. On top of this, we found a strong tendency to show, not tell sensations, and an impoverished language especially for non-experts. Without a shared reference point of physically feeling the sensation, we believe that design can be easier, especially when a language of design is underdeveloped.

Recently, online interfaces have emerged for sourcing content, distributing content, and distributing media itself. We found that moving to a web-based tool with Macaron greatly facillitated distribution. Online deployment simultaneously widens exposure and speeds development, making it easier for designers to be inspired by or directly build upon one another's work. The trend will accelerate as the field matures, but there will need to be concurrent development of accessible hardware to connect with software tools.

Crowdsourcing and broadcasting:

We reviewed some of the substantial challenges and spoke of one type of solution. In HapTurk (Chapter 6), we showed proxies of high quality haptic experiences can communicate key aspects on more shareable media, to gain access to crowdsource evaluation tools like Mechanical Turk. Proxies might be able to generalize in other ways, for example, using video to infer feedback about physically moving objects like the CuddleBits (*e.g.*, as in [172]).

Proxies can do more than elicit feedback from crowds. It might also be a viable way to translate sensations between representations, analogous to downsampling high-definition video broadcasts for standard definition televisions. More exotic proxies, like using vibrotactile feedback to represent force or friction could be considered. Future perceptual studies are needed to accomplish this.

Open Haptics – design sharing communities:

Other kinds of designer-facing online communities and outreach activities may assist with open haptic media – making it easier to share design resources, build up a haptics design culture, and, where possible, cooperate on establishing a consistent design language of haptics. For example, online software like VibViz [**?**] and Macaron (Chapter 5) provide details to designers anywhere, while open hardware projects like HapKit [168] and Haply [77] are available to hackers and students. Conference workshops and hardware kits provide users and designers with additional means to experiment with advanced haptics.

Each of these projects solve different problems and provide independent benefits. Online collaboration, and resources like articles on haptic perception or tutorials on how to effectively create haptics, will connect more designers, artists, developers, students, and hackers and help to build haptics into new user experiences.

Examples, demos, Show, don't tell

We found that demonstrations were critical for haptic designers in the wild. To help elicit requirements, customers were brought in to try out various demos. To persuade customers, actually feeling the haptics was critical, if not always enough. The Haptic Instrument showed evidence for deictic features - what do you think of "this", which facillitated direct demonstration without having to resort to indirect linguistic descriptions. In Macaron, exposing the underlying structure of examples lead to improved understanding of how to develop sensations and developed haptic idioms.

9.3.2 Towards a Mature HaXD Tool Suite

Soft Features

Repeatedly, participants asked for "soft features", associated with more polished tools. This includes features like copy and paste, undo and redo, saving and loading, grouping, looping, reduced delay, and high-fidelity rendering. We increasingly found, as we iteratively developed HaXD tools, that these features are more important than getting the right paradigm. As long as a designer is able to freely and accurately create, they can work around awkward design metaphors.

Low barrier, wide walls, high ceiling

A major goal of our work was to support creativity with haptic technology, and to support low barriers, wide walls, and a high ceiling [178, 206]. We've confirmed that this is critical for HaXD, and found ways of accomplishing it. As we mentioned, our participants found soft features essential, which helped to free them to make mistakes, explore various options, and provide both accessibility (low barriers) and refinement (high ceilings). For example, in Tactile Animation, participants found the animation objects easier for exploration and non-expert designers, while vector sensations offered more fine-tuned control. Similarly, we found the more flexible track-based paradigm used by Macaron to allow for various paths through the interface, and was generalizable to other sensations like simple 1-DOF robots.

Engineering useful features into mature tools

Both individual experiences in designing with haptics, and reference to tools in other modalities show the value of seamless access to many small but useful features – direct manipulation, undo/redo, copy/paste, selection and group manipulation, import/export to various formats, access to a library, etc.. Individually, many of these do not present major research or usability problems, but integrating them is another matter. There are at least two obvious approaches:

Additive: – Haptics authoring capability can be added to existing mature platforms focused on another modality – *e.g.*, to Photoshop (for graphics) or Premier (video). Force-feedback is already integrable into certain video game and virtual world environments (Unity, XNA), but this is only a small subset of possible haptic experiences.

Haptics-specific: – To truly optimize haptics capabilities, it may be worthwhile to invest serious development into a haptics-specific platform; or more likely, many attempts at such a thing focused on different categories of hardware (*e.g.*, tactile wearables versus 3D desktop force feedback environments) or application area, perhaps by extending initial low-level editors already available.

Researchers are good at pioneering individual steps and features, but integration is a significant development initiative which lies more in the province of businesses who stand to profit by the effort invested. Which task focus and development pipelines are to be supported, including multimodal design and integration, will be driven by application areas with urgent needs and a promising economic prognosis:

e.g., for movie special effects versus surgical simulations or end-user customizations.

9.3.3 Notes on Implementing HaXD Tools

During our in-depth studies, we looked at three different platforms for implementing HaXD support tools: Android on tablet for mHIVE, Python on desktop for Mango, and JavaScript deployed on the web for Macaron. From these three implementations, and from attempting to distribute or deploy them, we have found several lessons for implementing interactive software for HaXD support. Many of these are straightforward applications of software engineering, but we particularly found recently developed web tools to be especially useful when managing the complex interfaces required by design tools.

Observer Pattern – Well established as a software engineering design pattern, we found the observer pattern was critical in keeping the complex interfaces of our design tools synchronized. This is especially critical because we need to synchronize haptic, audio, and video playback, and allow implementations through both temporal and spatial means. In Mango, we implemented this pattern ourselves, but in Macaron, we used React [] which automates the process. This latter approach was more efficient and facillitated programming.

Components and Track-based Editing – Another valuable approach to develop extensible tools was React's Components, composable interface elements that can be parameterized and reused. This technique, along with a carefully organized architecture and the generic track-based editor control paradigm used by Macaron made it very easy to extend. When developing MacaronBit by forking the original code for Macaron, we were able to repurpose the tool to control the CuddleBits in a single day. While a better paradigm could be designed, our in-lab experts were able to used generalized tracks to create differentiable behaviours from the bit. One can also imagine redesigning a track-based editor to use a different control paradigm with the same device - we do this with MacaronMix, which mixes two VT icons using again a fork off of Macaron.

Flexible, extensible data structures – Because of the complexity and variance of haptic devices, we needed to establish data structures that were lightweight but extensible. We used JavaScript Object Notation (JSON) to accomplish this in Mango, Macaron, and FeelCraft. JSON which allowed us to specify only the required information needed, whether for device parameters in Mango, or sensations created by both Mango and Macaron. Our tools would check JSON files for designed features - like amplitude tracks in Mango - and ignore others we left in for a stub - like frequency tracks. These structures were simple, human-readable, ignored extraneous detail, and could gracefully fail when poorly formatted. Because we need to accommodate different paradigms in HaXD tools, lightweight data types make a lot of sense.

Web Deployment – One challenge with haptic technology Web - more people, online communities, resources, show internal structure Need to package up software Documentation for use

No Delay – In interviews with designers, we found that delay was critical; multimodal sensations had to be tightly synchronized. We found similar problems with our tools. mHIVE had a short delay which was distracting to participants, particularly when they tried to play very short pulses. Macaron's real-time audio synthesizer sometimes created a "muddy" effect because it had a limited update rate. We found this disappeared after we implemented a .WAV exporting function. Combined with our findings that users would switch between focused, in-depth editing and more reflective experiencing, we suggest (and are working to implement) a pre-feel/render model, like with visual effects tools. This is one potential problem with current web implementations of our design tools, especially when attempting to communicate to embodied devices like Arduino.

Distribution requires hardware – An obvious, but critical, requirement for HaXD tools is that they require the appropriate hardware. Above we've discussed flexible architectures and data structures to support different hardwares and paradigms of sensations. In addition, potential designers need easy access to hardware. One way we've found is to create do-it-yourself haptic wristbands for Macaron. (URL) Critical for VT feedback is that amplifiers and actuators matter - we found the DIY custom amplifier we've built does not have the same crispness as a more expensive commercial audio amplifier. This means that modular systems could be designed for facillitate accessible haptic feedback and design, but are complicated in because the setup doesn't end at the actuator. Another approach is to use a HapTurk-based translation with mobile vibrations like those available on Android (link to hapticJS), although this impoverishes sensations and leads to the challenge of native platform, discussed above.

Logging and Analytics – One important feature for modern software is remote analytics. Various types of metrics and usage statistics help developers prioritize development, fix bugs, and report to investors. Interestingly, because haptic technology extends into the physical world, one can consider using remote analytics both on their design and playback software, and on the physical devices. With a modular system Logging and analytics not a problem; should record these features (from Macaron, a little Tactile Animation)

9.3.4 Evaluating HaXD Tools

Post interviews – Early in our investigation, we found that participants found traditional think-aloud protocols challenging. This appeared to be due to a split attention of paying close attention to sensory input on their hand while simultaneously designing sensations. Even without thinking-aloud, this was cognitively challenging, and participants wanted looping, or would step back to refine and feel their sensations. In response, we adopted unintrusive logging and post-interviews, both of which were effective in capturing aspects of the haptic design process.

Phenomenology – Because of the close connection with participant experience, we adopted techniques from phenomenology, both to capture the sensory experience of haptic sensations, and to capture the experience of design. This perspective was useful for rigorous examination of our participants' interviews and to guide the interview process. In addition, the practical techniques, considering each meaning unit and clustering them into both in-vivo language and analytical language, very useful, especially when later combined with Grounded Theory techniques.

Grounded Theory – To better equip ourselves to analyze larger sets of data and develop a broader theory, we also drew upon Grounded Theory as described by Corbin and Strauss [44]. Especially when analyzing screen-captured video, coding techniques were valuable in identifying countable data, strategies, and adding a second layer of data analysis techniques to complement those provided by phenomenology.

Analytics and Logs – Qualitative analysis of post-interviews provided valuable insight, but only gave a partial view of haptic experience design. Timing analysis, and visualized logs, captured data that may not have been not have been observed by the researcher, and not reported or noticed by the participant in a post-interview. Importantly, it also offers a transition into broader data collection at large, providing a sustainable method of observing features

Multimodal Design Tasks – Throughout our three in-depth design tool studies, we refined our approach for participant tasks. With the initial Haptic Instrument study, we asked participants to freely explore the interface and attempt to design for affective words like happy or sad. This was challenging for the participants, and we found inconsistent designs. ⌈**OS** *TODO: what exactly was bad about this?*⌋ In our Tactile Animation study, we used more defined scenarios - a heartbeat, a directional cue for driving, and a sound-based design task. The first two were effective and descriptive for our participants, but the sound task in particular engaged them; it also gave us controls over theoretical dimensions like complexity or abstractness, impose constraints like careful timing, and was externally valid with the multimodal nature of haptic design. In our final in-depth study, we adopted animations instead of sounds, with similar results.

Ratings – In our studies, we had participants rate their designs including confidence and difficulty. We did not find quantitative results from these reports, mostly due to low sample size and the qualitative nature of analysis. However, we did find it to be useful for two reasons. First, it provided an elicitation device, inviting participants to comment on the difficulty or their happiness with their designs. Second, as we scale to more natural settings and higher sample sizes, we hope quantitative findings from rating scales will provide insight to questions about challenge, learning, and flow theory, which emerged from the third in-depth study.

9.4 Future Work

The next steps for this work are to directly follow up on our synthesized recommendations to establish HaXD, continue to expand our our tools and evaluation methods, and invert the premise of this thesis: use haptic technology to support design and problem solving.

We are already following up on our synthesized recommendations. Macaron was deployed online, along with a preliminary DIY haptic wristband and amplifier[1] to enable anyone, anywhere to start prototyping vibrotactile icons. We are planning to develop 3D printed models and online distribution methods to further facilitate artists, designers, and makers to work with haptic technology. We are also reaching out to other haptic researchers to find a way to connect their work with end-users through tutorials, articles, and other online tools.

Next, we plan to expand our tools to support more paradigms, haptic sensations, and evaluation methods, primarily using Macaron as a development platform. We are actively developing MacaronMix, an extension of Macaron which support interpolation of VT icons using novel algorithms; this new tool paradigm takes two *browsed* examples and directly combines them. We also seek to expand Macaron's output modalities to include directional capabilities, such as that explored by tactile animation, and to further develop the user logs from Chapter 5 into an analytical suite to further capture the haptic design process. More generally, we plan to follow up on our findings on Csikszentmihalyi's theory of flow [49] as a means to evaluate creativity-support tools.

Finally, haptic technology can support design, problem-solving, and sense-making. Learning benefits from physically embodied interactions [170]; meanwhile, professionals like architects, designers, and psychiatrists approach each new problem or patient with fresh eyes, seeking to learn about it [198]. We believe that the link between haptic technology and creativity runs both ways. Possibilities in-

[1] http://www.instructables.com/id/MacaronKit-USB-Powered-Mono-Audio-Amplifier

clude developing accessible sense-making platforms [214], and exploring creative, haptic therapy to promote mental wellness.

9.5 In Closing

Our technology continues to push us toward a mixed reality, where physical objects compute, and people can enter realistic virtual worlds. We strive towards computers that speak on human terms, both in end-user experiences by engaging all of our senses, and by supporting human creativity when designing those experiences. ⌈**OS** *poignant final line needed*⌋

Bibliography

[1] M. Abdur Rahman, A. Alkhaldi, J. Cha, and A. El Saddik. Adding haptic feature to YouTube. In *Proceedings of the international conference on Multimedia - MM '10*, pages 1643–1646, New York, New York, USA, oct 2010. ACM Press. ISBN 9781605589336. doi:10.1145/1873951.1874310. URL http://dl.acm.org/citation.cfm?id=1873951.1874310. → pages 2, 17, 84

[2] J. Allen, L. Cang, M. Phan-Ba, A. Strang, and K. MacLean. Introducing the Cuddlebot: A Robot That Responds to Touch Gestures. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, HRI'15 Extended Abstracts, page 295, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3318-4. doi:10.1145/2701973.2702698. URL http://doi.acm.org/10.1145/2701973.2702698. → pages 10, 172

[3] D. Alles. Information Transmission by Phantom Sensations. *IEEE Transactions on Man Machine Systems*, 11(1):85–91, Mar. 1970. ISSN 0536-1540. doi:10.1109/TMMS.1970.299967. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4081935. → pages 10, 40, 41, 46, 50

[4] Android Open Source Project. Android Developers, 2012. URL http://developer.android.com/index.html. → pages 30

[5] F. Arab, S. Paneels, M. Anastassova, S. Coeugnet, F. Le Morellec, A. Dommes, and A. Chevalier. Haptic patterns and older adults: To repeat or not to repeat? In *2015 IEEE World Haptics Conference (WHC)*, pages 248–253. IEEE, jun 2015. ISBN 978-1-4799-6624-0. doi:10.1109/WHC.2015.7177721. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7177721. → pages 8, 14, 80, 82

[6] I. Asimov. *Foundation and Empire*. Gnome Press, 1952. → pages 27

[7] M. Azmandian, M. Hancock, H. Benko, E. Ofek, and A. D. Wilson. Haptic Retargeting: Dynamic repurposing of passive haptics for enhanced virtual reality experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 1968–1979, New York, New York, USA, 2016. ACM Press. ISBN 9781450333627. doi:10.1145/2858036.2858226. URL http://dl.acm.org/citation.cfm?doid=2858036.2858226. → pages 10, 12

[8] P. Bach-y Rita and Kaczmarek. Form perception with a 49-point electrotactile stimulus array on the tongue: A technical note. *Journal of rehabilitation research and development*, 35(4):427, jan 1998. ISSN 0748-7711. → pages 9

[9] P. Bach-y Rita, C. C. Collins, F. A. Saunders, B. White, and L. Scadden. Vision Substitution by Tactile Image Projection. *Nature*, 221(5184): 963–964, mar 1969. ISSN 0028-0836. doi:10.1038/221963a0. URL http://dx.doi.org/10.1038/221963a0. → pages 11

[10] O. Bau, I. Poupyrev, A. Israr, and C. Harrison. TeslaTouch: Electrovibration for Touch Surfaces. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology - UIST '10*, page 283, New York, New York, USA, oct 2010. ACM Press. ISBN 9781450302715. doi:10.1145/1866029.1866074. URL http://dl.acm.org/citation.cfm?id=1866029.1866074. → pages 8

[11] J. Bliss, M. Katcher, C. Rogers, and R. Shepard. Optical-to-Tactile Image Conversion for the Blind. *IEEE Transactions on Man Machine Systems*, 11 (1):58–65, mar 1970. ISSN 0536-1540. doi:10.1109/TMMS.1970.299963. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4081931. → pages 8, 11, 130

[12] B. S. Bloom and M. D. Engelhart. *Taxonomy of Educational Objectives: The Classification of Educational Goals: By a Committee of College and University Examiners: Handbook 1*. David McKay, 1969. → pages xv, 120, 121

[13] J. R. Blum, I. Frissen, and J. R. Cooperstock. Improving Haptic Feedback on Wearable Devices through Accelerometer Measurements. In *Proceedings of the 28th Annual ACM Symposium on User Interface*

Software & Technology - UIST '15, pages 31–36, New York, New York, USA, nov 2015. ACM Press. ISBN 9781450337793. doi:10.1145/2807442.2807474. URL http://dl.acm.org/citation.cfm?id=2807442.2807474. → pages 10, 178

[14] R. A. Bolt. Put-that-there: Voice and Gesture at the Graphics Interface. *ACM SIGGRAPH Computer Graphics*, 14(3):262–270, July 1980. ISSN 00978930. doi:10.1145/965105.807503. URL http://dl.acm.org/citation.cfm?id=965105.807503. → pages 33

[15] L. Bonanni, C. Vaucelle, J. Lieberman, and O. Zuckerman. TapTap: A Haptic Wearable for Asynchronous Distributed Touch Therapy. In *CHI '06 extended abstracts on Human factors in computing systems - CHI EA '06*, page 580, New York, New York, USA, apr 2006. ACM Press. ISBN 1595932984. doi:10.1145/1125451.1125573. URL http://dl.acm.org/citation.cfm?id=1125451.1125573. → pages 13

[16] S. Brave and A. Dahley. inTouch: a medium for haptic interpersonal communication. In *CHI '97 extended abstracts on Human factors in computing systems looking to the future - CHI '97*, pages 363–364, New York, NY, USA, 1997. ACM Press. ISBN 0897919262. doi:10.1145/1120212.1120435. URL http://portal.acm.org/citation.cfm?doid=1120212.1120435. → pages 13

[17] S. Brewster and L. M. Brown. Tactons: structured tactile messages for non-visual information display. In *AUIC '04*, pages 15–23, Jan. 2004. URL http://dl.acm.org/citation.cfm?id=976310.976313. → pages 19, 26, 29, 63, 80, 108, 110

[18] S. Brewster and A. Constantin. Tactile feedback for ambient awareness in mobile interactions. In *BCS '10*, pages 412–417. British Computer Society, Sept. 2010. ISBN 978-1-78017-130-2. URL http://dl.acm.org/citation.cfm?id=2146303.2146365. → pages 14

[19] S. A. Brewster, P. C. Wright, and A. D. N. Edwards. An evaluation of earcons for use in auditory human-computer interfaces. In *CHI '93)*, pages 222–227, New York, USA, May 1993. ACM Press. ISBN 0897915755. doi:10.1145/169059.169179. URL http://dl.acm.org/citation.cfm?id=169059.169179. → pages 19, 26

[20] L. Brown, S. Brewster, and H. Purchase. A First Investigation into the Effectiveness of Tactons. In *World Haptics '05*, pages 167–176. IEEE,

2005. ISBN 0-7695-2310-2. doi:10.1109/WHC.2005.6. URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1406930. → pages 19, 26

[21] L. M. Brown and T. Kaaresoja. Feel who's talking: Using Tactons for Mobile Phone Alerts. In *CHI '06 extended abstracts on Human factors in computing systems - CHI EA '06*, page 604, New York, New York, USA, apr 2006. ACM Press. ISBN 1595932984. doi:10.1145/1125451.1125577. URL http://dl.acm.org/citation.cfm?doid=1125451.1125577http://dl.acm.org/citation.cfm?id=1125451.1125577. → pages 14

[22] L. M. Brown, S. A. Brewster, and H. C. Purchase. Tactile crescendos and sforzandos. In *CHI Extened Abstracts '06*, pages 610–615, New York, USA, Apr. 2006. ACM Press. ISBN 1595932984. doi:10.1145/1125451.1125578. URL http://dl.acm.org/citation.cfm?id=1125451.1125578. → pages 14, 19, 26, 29, 63

[23] L. M. Brown, S. A. Brewster, and H. C. Purchase. Multidimensional tactons for non-visual information presentation in mobile devices. In *MobileHCI '06*, pages 231–238, New York, USA, Sept. 2006. ACM Press. ISBN 1595933905. doi:10.1145/1152215.1152265. URL http://dl.acm.org/citation.cfm?id=1152215.1152265. → pages 19, 26, 29, 80, 82

[24] L. Brunet, C. Megard, S. Paneels, G. Changeon, J. Lozada, M. P. Daniel, and F. Darses. Invitation to the voyage: The design of tactile metaphors to fulfill occasional travelers' needs in transportation networks. In *2013 World Haptics Conference (WHC)*, pages 259–264. IEEE, apr 2013. ISBN 978-1-4799-0088-6. doi:10.1109/WHC.2013.6548418. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6548418. → pages 14, 80, 82, 179

[25] P. Bucci, X. L. Cang, M. Chun, D. Marino, O. Schneider, H. Seifi, and K. MacLean. CuddleBits: an iterative prototyping platform for complex haptic display. In *EuroHaptics '16 Demos*, 2016. → pages vi, 102

[26] P. Buttolo, R. Oboe, and B. Hannaford. Architectures for shared haptic virtual environments. *Computers & Graphics*, 21(4):421–429, jul 1997. ISSN 00978493. doi:10.1016/S0097-8493(97)00019-8. URL http://dx.doi.org/10.1016/S0097-8493(97)00019-8. → pages 9

[27] B. Buxton. *Sketching User Experiences: Getting the Design Right and the Right Design*. Morgan Kaufmann Publishers Inc., 2007. → pages xvii, 3, 15, 16, 62, 65, 134, 140, 175

[28] G. A. Calvert, M. J. Brammer, and S. D. Iversen. Crossmodal identification. *Trends in Cognitive Sciences*, 2(7):247–253, July 1998. ISSN 1364-6613. doi:10.1016/S1364-6613(98)01189-9. → pages 115

[29] G. Campion. The Pantograph Mk-II: A Haptic Instrument. pages 45–58. Springer London, 2005. doi:10.1007/978-0-85729-576-7_3. URL http://link.springer.com/10.1007/978-0-85729-576-7{_}3. → pages 9

[30] L. Cang, P. Bucci, and K. E. MacLean. CuddleBits: Friendly, Low-cost Furballs that Respond to Touch. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 365–366. ACM, 2015. → pages 10, 126

[31] K. J. Carbonneau, S. C. Marley, and J. P. Selig. A meta-analysis of the efficacy of teaching mathematics with concrete manipulatives. *Journal of Educational Psychology*, 105(2):380–400, 2013. ISSN 1939-2176. doi:10.1037/a0031084. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/a0031084. → pages 14, 115

[32] T. Carter, S. Seah, B. Long, and B. Drinkwater. UltraHaptics: multi-point mid-air haptic feedback for touch surfaces. *Proceedings of the 26th*, 2013. URL http://dl.acm.org/citation.cfm?id=2502018. → pages 9

[33] J. R. Cauchard, J. L. Cheng, T. Pietrzak, and J. A. Landay. ActiVibe: Design and Evaluation of Vibrations for Progress Monitoring. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 3261–3271, New York, New York, USA, 2016. ACM Press. ISBN 9781450333627. doi:10.1145/2858036.2858046. URL http://dl.acm.org/citation.cfm?doid=2858036.2858046. → pages 10, 14

[34] J. Cha, Y.-S. Ho, Y. Kim, J. Ryu, and I. Oakley. A Framework for Haptic Broadcasting. *IEEE Multimedia*, 16(3):16–27, July 2009. ISSN 1070-986X. doi:10.1109/MMUL.2009.42. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5228689. → pages 19

[35] A. Chan, K. MacLean, and J. McGrenere. Designing haptic icons to support collaborative turn-taking. *International Journal of Human-Computer Studies*, 66(6):333–355, 2008. URL

http://www.sciencedirect.com/science/article/pii/S1071581907001565. → pages 11, 13, 19, 26, 82, 84, 86, 87

[36] A. Chang and C. O'Sullivan. Audio-haptic feedback in mobile phones. In *CHI EA '05*, pages 1264–1267, Apr. 2005. ISBN 1595930027. doi:10.1145/1056808.1056892. URL http://dl.acm.org/citation.cfm?id=1056808.1056892. → pages 18, 103

[37] A. Chang, S. O'Modhrain, R. Jacob, E. Gunther, and H. Ishii. ComTouch: design of a vibrotactile communication device. In *Proceedings of the conference on Designing interactive systems processes, practices, methods, and techniques - DIS '02*, pages 312–320, New York, NY, USA, 2002. ACM Press. ISBN 1581135157. doi:10.1145/778712.778755. URL http://portal.acm.org/citation.cfm?doid=778712.778755. → pages 13

[38] J. Chang, K. MacLean, and S. Yohanan. Gesture recognition in the haptic creature. In *Proceedings of the 2010 international conference on Haptics: generating and perceiving tangible sensations, Part I*, EuroHaptics'10, pages 385–391, Berlin, Heidelberg, 2010. Springer-Verlag. ISBN 3-642-14063-7, 978-3-642-14063-1. URL http://portal.acm.org/citation.cfm?id=1884164.1884223. → pages 172

[39] G. Changeon, D. Graeff, M. Anastassova, and J. Lozada. Tactile Emotions: A Vibrotactile Tactile Gamepad for Transmitting Emotional Messages to Children with Autism. pages 79–90. Springer Berlin Heidelberg, 2012. doi:10.1007/978-3-642-31401-8_8. URL http://link.springer.com/10.1007/978-3-642-31401-8{_}8. → pages 13

[40] A. Charpentier. *Analyse experimentale de quelques elements de la sensation de poids*. 1891. → pages 10, 135

[41] L.-P. Cheng, P. Lühne, P. Lopes, C. Sterz, and P. Baudisch. Haptic turk: a motion platform based on people. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 3463–3472, New York, New York, USA, apr 2014. ACM Press. ISBN 9781450324731. doi:10.1145/2556288.2557101. URL http://dl.acm.org/citation.cfm?id=2556288.2557101. → pages 12

[42] L.-P. Cheng, T. Roumen, H. Rantzsch, S. Köhler, P. Schmidt, R. Kovacs, J. Jasper, J. Kemper, and P. Baudisch. TurkDeck. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15*, pages 417–426, New York, New York, USA, 2015. ACM Press.

ISBN 9781450337793. doi:10.1145/2807442.2807463. URL
http://dl.acm.org/citation.cfm?doid=2807442.2807463. → pages 12

[43] S. Choi and K. J. Kuchenbecker. Vibrotactile Display: Perception, Technology, and Applications. *Proceedings of the IEEE*, 101(9): 2093–2104, sep 2013. ISSN 0018-9219. doi:10.1109/JPROC.2012.2221071. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6353870. → pages 2, 8, 135, 141

[44] J. Corbin and A. Strauss. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Sage Publications, Inc., 3 edition, 2008. → pages 20, 32, 54, 68, 120, 133, 138, 186

[45] J. W. Creswell. *Qualitative inquiry and research design: choosing among five approaches*. Sage Publications Inc, 3rd edition, 2013. → pages 20, 27, 31, 32

[46] N. Cross. *Designerly Ways of Knowing*. Springer-Verlag London Limited, 2006. → pages 16, 134

[47] N. Cross. Creative thinking in design. In *Proceedings of the 2007 Symposium on Science of Design - SoD '07*, page 2, New York, New York, USA, Mar. 2007. ACM Press. ISBN 9781605584362. doi:10.1145/1496630.1496632. URL http://dl.acm.org/citation.cfm?id=1496630.1496632. → pages 21

[48] N. Cross. *Design Thinking: Understanding How Designers Think and Work*. Berg Publishers, Oxford, UK, 2011. → pages 21, 36

[49] M. Csikszentmihalyi. *Creativity: Flow and the Psychology of Discovery and Invention*. HarperCollins Publishers, Inc., 1st edition, 1996. → pages 64, 73, 134, 187

[50] D. Cuartielles, A. Goransson, T. Olsson, and S. Stenslie. Developing Visual Editors for High-Resolution Haptic Patterns. In *HAID'12 Posters and Demos*, pages 42–44, 2012. URL http://www.english.certec.lth.se/haptics/HAID12/-proceedings.pdf. → pages 18, 43, 44

[51] H. Culbertson, J. Unwin, and K. J. Kuchenbecker. Modeling and rendering realistic textures from unconstrained tool-surface interactions. *IEEE Transactions on Haptics*, 7(3):381–93, Jan. 2014. ISSN 2329-4051.

doi:10.1109/TOH.2014.2316797. URL
http://www.ncbi.nlm.nih.gov/pubmed/25248220. → pages 8, 18, 62, 137

[52] F. Danieau, J. Fleureau, A. Cabec, P. Kerbiriou, P. Guillotel, N. Mollet, M. Christie, and A. Lecuyer. Framework for enhancing video viewing experience with haptic effects of motion. In *2012 IEEE Haptics Symposium (HAPTICS)*, pages 541–546. IEEE, mar 2012. ISBN 978-1-4673-0809-0. doi:10.1109/HAPTIC.2012.6183844. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6183844. → pages 17

[53] F. Danieau, J. Fleureau, P. Guillotel, N. Mollet, A. Lécuyer, and M. Christie. HapSeat: producing motion sensation with multiple force-feedback devices embedded in a seat. In *VRST '12*, pages 69–76, New York, New York, USA, Dec. 2012. ACM Press. ISBN 9781450314695. doi:10.1145/2407336.2407350. URL http://dl.acm.org/citation.cfm?id=2407336.2407350. → pages 17, 40, 57

[54] F. Danieau, J. Bernon, J. Fleureau, P. Guillotel, N. Mollet, M. Christie, and A. Lécuyer. H-Studio: An authoring tool for adding haptic and motion effects to audiovisual content. In *Proceedings of the adjunct publication of the 26th annual ACM symposium on User interface software and technology - UIST '13 Adjunct*, pages 83–84, New York, New York, USA, oct 2013. ACM Press. ISBN 9781450324069. doi:10.1145/2508468.2514721. URL http://dl.acm.org/citation.cfm?id=2508468.2514721. → pages 17

[55] F. Danieau, J. Fleureau, P. Guillotel, N. Mollet, M. Christie, and A. Lécuyer. HapSeat: a novel approach to simulate motion in audiovisual experiences. In *ACM SIGGRAPH 2013*, pages 3035–3038, Anaheim, CA, USA, apr 2013. ACM Press. ISBN 9781450319522. doi:10.1145/2468356.2479604. URL http://dl.acm.org/citation.cfm?id=2468356.2479604http://dl.acm.org/citation.cfm?id=2503368.2503374. → pages 17

[56] F. Danieau, A. Lecuyer, P. Guillotel, J. Fleureau, N. Mollet, and M. Christie. Enhancing audiovisual experience with haptic feedback: a survey on HAV. *IEEE transactions on haptics*, 6(2):193–205, jan 2013. ISSN 2329-4051. doi:10.1109/TOH.2012.70. URL http://www.ncbi.nlm.nih.gov/pubmed/24808303. → pages 13, 17

[57] 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*, 21(2): 11–21, 2014. ISSN 1070-986X. doi:10.1109/MMUL.2013.64. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6695749. → pages 18, 19, 103, 136

[58] M. Dingemanse. Advances in the Cross-Linguistic Study of Ideophones. *Language and Linguistics Compass*, 6(10):654–672, oct 2012. ISSN 1749818X. doi:10.1002/lnc3.361. URL http://doi.wiley.com/10.1002/lnc3.361. → pages 126

[59] P. Dourish. *Where the action is: the foundations of embodied interaction*. MIT Press, 2004. → pages 115

[60] S. Dow, J. Fortuna, D. Schwartz, B. Altringer, D. Schwartz, and S. Klemmer. Prototyping dynamics: sharing multiple designs improves exploration, group rapport, and results. In *CHI '11*, pages 2807–2816. ACM Press, May 2011. ISBN 9781450302289. doi:10.1145/1978942.1979359. URL http://dl.acm.org/citation.cfm?id=1978942.1979359. → pages 16, 21, 62, 74

[61] M. Eid, A. Alamri, and A. Saddik. MPEG-7 Description of Haptic Applications Using HAML. In *2006 IEEE International Workshop on Haptic Audio Visual Environments and their Applications (HAVE 2006)*, pages 134–139. IEEE, 2006. ISBN 1-4244-0761-3. doi:10.1109/HAVE.2006.283780. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4062526. → pages 17, 84

[62] M. Eid, S. Andrews, A. Alamri, and A. E. Saddik. HAMLAT: A HAML-Based Authoring Tool for Haptic Application Development. In *LNCS 5024 - Haptics: Perception, Devices and Scenarios*, volume 5024, pages 857–866. 2008. ISBN 978-3-540-69056-6. doi:10.1007/978-3-540-69057-3. URL http://www.springerlink.com/index/10.1007/978-3-540-69057-3. → pages 17, 44, 84

[63] Z. Eitan and I. Rothschild. How music touches: Musical parameters and listeners' audio-tactile metaphorical mappings. *Psychology of Music*, 39 (4):449–467, Nov. 2010. ISSN 0305-7356.

doi:10.1177/0305735610377592. URL
http://pom.sagepub.com/content/39/4/449.short. → pages 19, 26

[64] P. Ekman and W. V. Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124–9, feb 1971. ISSN 0022-3514. URL http://www.ncbi.nlm.nih.gov/pubmed/5542557. → pages 11

[65] P. Ekman, I. R. Davidson, P. Ellsworth, and W. V. Friesen. Are There Basic Emotions? *Psychological Review*, 99(3):550–553, 1992. → pages 11, 31

[66] C. A. Ellis, S. J. Gibbs, and G. Rein. Groupware: some issues and experiences. *Communications of the ACM*, 34(1):39–58, Jan. 1991. ISSN 00010782. doi:10.1145/99977.99987. URL http://dl.acm.org/citation.cfm?id=99977.99987. → pages 16, 135

[67] M. Enriquez and K. MacLean. The hapticon editor: a tool in support of haptic communication research. In *HAPTICS '03*, pages 356–362. IEEE Comput. Soc, 2003. ISBN 0-7695-1890-7. doi:10.1109/HAPTIC.2003.1191310. URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1191310. → pages 9, 17, 19, 24, 26, 36, 40, 42, 43, 48, 64, 65, 137

[68] M. Enriquez, K. MacLean, and C. Chita. Haptic phonemes: basic building blocks of haptic communication. In *ICMI '06*, page 302, New York, USA, Nov. 2006. ACM Press. ISBN 159593541X. doi:10.1145/1180995.1181053. URL http://dl.acm.org/citation.cfm?id=1180995.1181053. → pages 19, 26, 64

[69] D. Feygin, M. Keehner, and R. Tendick. Haptic guidance: experimental evaluation of a haptic training method for a perceptual motor skill. In *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, pages 40–47. IEEE Comput. Soc, 2002. ISBN 0-7695-1489-8. doi:10.1109/HAPTIC.2002.998939. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=998939. → pages 15

[70] A. Flagg and K. MacLean. Affective touch gesture recognition for a furry zoomorphic machine. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction - TEI '13*, page 25, New York, New York, USA, 2013. ACM Press. ISBN 9781450318983.

doi:10.1145/2460625.2460629. URL
http://dl.acm.org/citation.cfm?doid=2460625.2460629. → pages 12

[71] A. Flagg, D. Tam, K. MacLean, and R. Flagg. Conductive fur sensing for a gesture-aware furry robot. In *2012 IEEE Haptics Symposium (HAPTICS)*, pages 99–104. IEEE, mar 2012. ISBN 978-1-4673-0809-0. doi:10.1109/HAPTIC.2012.6183776. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6183776. → pages 12

[72] N. Fleming and C. Mills. Not another inventory, rather a catalyst for reflection. *To Improve the Academy*, 11, 1992. → pages 120

[73] FMOD. FMOD Ex API, 2013. URL http://www.fmod.org. → pages 30

[74] J. Forsslund, M. Yip, and E.-L. Sallnäs. WoodenHaptics. In *TEI '14*, pages 133–140. ACM Press, Jan. 2015. ISBN 9781450333054. doi:10.1145/2677199.2680595. URL http://dl.acm.org/citation.cfm?id=2677199.2680595. → pages 9, 18, 136

[75] Futurama. Parasites Lost, 2001. → pages 27

[76] A. Gallace. The science of interpersonal touch: An overview. *Neuroscience & Biobehavioral Reviews*, 34(2):246–259, 2010. ISSN 01497634. doi:10.1016/j.neubiorev.2008.10.004. → pages 13

[77] C. Gallacher, A. Mohtat, and S. Ding. Toward Open-Source Portable Haptic Displays with Visual-Force-Tactile Feedback Colocation. In *HAPTICS '16*, 2016. ISBN 9781509009039. → pages 9, 136, 182

[78] Y. Gao, H. A. Osman, and A. El Saddik. MPEG-V based web haptic authoring tool. In *IEEE International Symposium on Haptic Audio Visual Environments and Games (HAVE)*, pages 87–91. IEEE, oct 2013. ISBN 978-1-4799-0849-3. doi:10.1109/HAVE.2013.6679616. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6679616. → pages 13, 17

[79] D. Gaw, D. Morris, and K. Salisbury. Haptically Annotated Movies: Reaching Out and Touching the Silver Screen. In *2006 14th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 287–288. IEEE, 2006. ISBN 1-4244-0226-3. doi:10.1109/HAPTIC.2006.1627106. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1627106. → pages 13

[80] S. Gerofsky. Mathematical learning and gesture: Character viewpoint and observer viewpoint in students' gestured graphs of functions. *Gesture*, 10 (2-3):321–343, 2010. → pages 14

[81] B. G. Glaser and A. L. Strauss. *Awareness of dying*. Transaction Publishers, 1966. → pages 20

[82] A. M. Glenberg, T. Gutierrez, J. R. Levin, S. Japuntich, and M. P. Kaschak. Activity and Imagined Activity Can Enhance Young Children's Reading Comprehension. *Journal of Educational Psychology*, 96:424, 2004. → pages 115

[83] S. Greenberg and C. Fitchett. Phidgets: Easy development of physical interfaces through physical widgets. In *UIST '01*, page 209, New York, New York, USA, Nov. 2001. ACM Press. ISBN 158113438X. doi:10.1145/502348.502388. URL http://dl.acm.org/citation.cfm?id=502348.502388. → pages 18, 136

[84] M. Grunwald and M. John. German pioneers of research into human haptic perception. In *Human Haptic Perception: Basics and Applications*, pages 15–39. Birkhäuser Basel, Basel, 2008. doi:10.1007/978-3-7643-7612-3_2. URL http://www.springerlink.com/index/10.1007/978-3-7643-7612-3{_}2. → pages 7

[85] S. Guest, J. M. Dessirier, A. Mehrabyan, F. McGlone, G. Essick, G. Gescheider, A. Fontana, R. Xiong, R. Ackerley, and K. Blot. The development and validation of sensory and emotional scales of touch perception. *Attention, perception & psychophysics*, 73(2):531–50, Feb. 2011. ISSN 1943-393X. doi:10.3758/s13414-010-0037-y. URL http://www.ncbi.nlm.nih.gov/pubmed/21264727. → pages 82, 83

[86] P. Guillotel, F. Danieau, J. Fleureau, and I. Rouxel. Introducing Basic Principles of Haptic Cinematography Editing. In *Eurographics Workshop on Intelligent Cinematography and Editing*, pages 1–7, 2016. → pages 13, 18, 136

[87] A. L. Guinan, M. N. Montandon, A. J. Doxon, and W. R. Provancher. Discrimination thresholds for communicating rotational inertia and torque using differential skin stretch feedback in virtual environments. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 277–282. IEEE, feb 2014. ISBN 978-1-4799-3131-6. doi:10.1109/HAPTICS.2014.6775467. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6775467. → pages 12

[88] E. Gunther, G. Davenport, and S. O'Modhrain. Cutaneous grooves: composing for the sense of touch. In *NIME '02*, pages 1–6, New York, NY, USA, May 2002. ISBN 1-87465365-8. URL http://dl.acm.org/citation.cfm?id=1085171.1085181. → pages 11, 17, 19, 26, 29, 30, 42, 63, 64, 136

[89] A. Gupta, T. Pietrzak, N. Roussel, and R. Balakrishnan. Direct Manipulation in Tactile Displays. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 3683–3693, 2016. doi:10.1145/2858036.2858161. URL http://doi.acm.org/10.1145/2858036.2858161. → pages 8, 10

[90] A. Haans and W. IJsselsteijn. Mediated social touch: a review of current research and future directions. *Virtual Reality*, 9(2-3):149–159, jan 2006. ISSN 1359-4338. doi:10.1007/s10055-005-0014-2. URL http://link.springer.com/10.1007/s10055-005-0014-2. → pages 13

[91] S. G. Hart and L. E. Staveland. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology*, 52(C):139–183, 1988. ISSN 01664115. doi:10.1016/S0166-4115(08)62386-9. → pages 119

[92] B. Hartmann, L. Yu, A. Allison, Y. Yang, and S. R. Klemmer. Design as Exploration : Creating Interface Alternatives through Parallel Authoring and Runtime Tuning. In *UIST*, pages 91–100, 2008. ISBN 9781595939753. → pages 15, 134

[93] V. Hayward. A brief taxonomy of tactile illusions and demonstrations that can be done in a hardware store. *Brain research bulletin*, 75(6):742–52, Apr. 2008. ISSN 0361-9230. doi:10.1016/j.brainresbull.2008.01.008. URL http://www.sciencedirect.com/science/article/pii/S0361923008000178. → pages 10, 19, 41, 43, 82

[94] V. Hayward. Tactile Illusions. In *Scholarpedia of Touch*, pages 327–342. Atlantis Press, Paris, 2016. doi:10.2991/978-94-6239-133-8_27. URL http://link.springer.com/10.2991/978-94-6239-133-8{_}27. → pages 10, 135

[95] V. Hayward and K. Maclean. Do it yourself haptics: part I. *IEEE Robotics & Automation Magazine*, 14(4):88–104, Dec. 2007. ISSN 1070-9932. doi:10.1109/M-RA.2007.907921. URL http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4437756. → pages 2, 7, 19

[96] F. Hemmert and G. Joost. Ambient Life: Calm and Excited Pulsation as a Means of Life-like Permanent Tactile Status Display in Mobile Phones. In P. Desmet, S. Tzvetanova, P. Hekkert, and L. Justice, editors, *Proceedings from the 6th Conference on Design & Emotion 2008*. School of Design, The Hong Kong Polytechnic University SAR, oct 2008. ISBN 978-988-17489-2-8. → pages 14

[97] S. R. Herring, C.-C. Chang, J. Krantzler, and B. P. Bailey. Getting inspired! Understanding How and Why Examples are Used in Creative Design Practice. In *CHI '09*, pages 87–96, New York, New York, USA, Apr. 2009. ACM Press. ISBN 9781605582467. doi:10.1145/1518701.1518717. URL http://dl.acm.org/citation.cfm?id=1518701.1518717. → pages 15, 16, 43, 62, 65, 134

[98] E. Hoggan, C. Stewart, L. Haverinen, G. Jacucci, and V. Lantz. Pressages: augmenting phone calls with non-verbal messages. pages 555–562, oct 2012. doi:10.1145/2380116.2380185. URL http://dl.acm.org/citation.cfm?id=2380116.2380185. → pages 13, 84

[99] K. Hong, J. Lee, and S. Choi. Demonstration-based vibrotactile pattern authoring. In *TEI '13*, page 219, New York, New York, USA, Feb. 2013. ACM Press. ISBN 9781450318983. doi:10.1145/2460625.2460660. URL http://dl.acm.org/citation.cfm?id=2460625.2460660. → pages 17, 42, 137

[100] E. Hutchins. *Cognition in the Wild*, volume 19. MIT Press, 1995. ISBN 0262082314. doi:10.1098/rsbl.2011.0352. URL http://www.ida.liu.se/~nilda/CST-papers/Hutchins.pdf. → pages 15, 19

[101] I. Hwang and S. Choi. Subject-specific distortions in haptic perception of force direction. In *EuroHaptics 2014, Part I, LNCS*, volume 8618, pages 48–232–240, 2014. ISBN 978-3-662-44192-3. doi:10.1007/978-3-662-44193-0. URL http://link.springer.com/10.1007/978-3-662-44193-0. → pages 18

[102] I. Hwang, J. Seo, M. Kim, and S. Choi. Vibrotactile Perceived Intensity for Mobile Devices as a Function of Direction, Amplitude, and Frequency. *IEEE Transactions on Haptics*, 6(3):352–362, July 2013. ISSN 1939-1412. doi:10.1109/TOH.2013.2. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6420831. → pages 80

[103] W. IJsselsteijn. Presence in the past: What can we learn from media history? In *Being There - Concepts, Effects and Measurements of User*

Presence in Synthetic Environments, pages 17–40. IOS Press, Amsterdam, 2003. → pages 12, 41

[104] A. Israr and I. Poupyrev. Exploring surround haptics displays. In *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems - CHI EA '10*, pages 4171–4176, New York, New York, USA, apr 2010. ACM Press. ISBN 9781605589305. doi:10.1145/1753846.1754121. URL http://dl.acm.org/citation.cfm?id=1753846.1754121. → pages 8

[105] A. Israr and I. Poupyrev. Tactile brush: drawing on skin with a tactile grid display. In *CHI '11*, pages 2019–2028, Vancouver, BC, May 2011. ACM Press. ISBN 9781450302289. doi:10.1145/1978942.1979235. URL http://dl.acm.org/citation.cfm?id=1978942.1979235. → pages 10, 40, 41, 42, 46, 48, 50, 64, 75, 82

[106] A. Israr, I. Poupyrev, C. Ioffreda, J. Cox, N. Gouveia, H. Bowles, A. Brakis, B. Knight, K. Mitchell, and T. Williams. Surround Haptics: Sending Shivers Down Your Spine. In *SIGGRAPH Emerging Technologies*, Vancouver, Canada, Aug. 2011. ISBN 9781450309691. doi:10.1145/2048259.2048273. URL http://dl.acm.org/citation.cfm?id=2048259.2048273. → pages 40, 41

[107] A. Israr, S.-C. Kim, J. Stec, and I. Poupyrev. Surround haptics: tactile feedback for immersive gaming experiences. In *CHI EA '12*, pages 1087–1090, New York, New York, USA, May 2012. ACM Press. ISBN 9781450310161. doi:10.1145/2212776.2212392. URL http://dl.acm.org/citation.cfm?id=2212776.2212392. → pages 8

[108] A. Israr, S. Zhao, K. Schwalje, R. Klatzky, and J. Lehman. Feel Effects: Enriching Storytelling with Haptic Feedback. *Transactions on Applied Perception (TAP)*, 11(3), 2014. → pages 18, 19, 62, 64, 80, 82, 102, 103, 104, 106, 107, 108, 137, 179, 180

[109] A. Israr, S. Zhao, and O. Schneider. Exploring Embedded Haptics for Social Networking and Interactions. In *CHI Extended Abstracts '15*, pages 1899–1904, New York, New York, USA, Apr. 2015. ACM Press. ISBN 9781450331463. doi:10.1145/2702613.2732814. URL http://dl.acm.org/citation.cfm?id=2702613.2732814. → pages v, 84, 87, 88, 101

[110] S. Jang, L. H. Kim, K. Tanner, H. Ishii, and S. Follmer. Haptic Edge Display for Mobile Tactile Interaction. In *Proceedings of the 2016 CHI*

Conference on Human Factors in Computing Systems - CHI '16, pages 3706–3716, New York, New York, USA, 2016. ACM Press. ISBN 9781450333627. doi:10.1145/2858036.2858264. URL http://dl.acm.org/citation.cfm?doid=2858036.2858264. → pages 9

[111] C. V. Jansson-Boyd. Touch matters: exploring the relationship between consumption and tactile interaction. *Social Semiotics*, 21(4):531–546, Sept. 2011. ISSN 1035-0330. doi:10.1080/10350330.2011.591996. URL http://dx.doi.org/10.1080/10350330.2011.591996. → pages 2, 11, 19, 27, 36, 80, 83, 130, 137

[112] J. Johnson and A. Henderson. Conceptual models: begin by designing what to design. *Interactions*, 9(1):25–32, Jan. 2002. ISSN 10725520. doi:10.1145/503355.503366. URL http://dl.acm.org/ft_gateway.cfm?id=503366&type=html. → pages 43

[113] J. Johnson and A. Henderson. Conceptual models: begin by designing what to design. *Interactions*, 9(1):25–32, 2002. → pages 116

[114] L. Jones. News from the Field: Courses in Haptics. *Transactions on Haptics*, 7(4):413–414, 2014. → pages 9, 19, 136, 165

[115] L. Jones, M. Nakamura, and B. Lockyer. Development of a tactile vest. In *HAPTICS '04*, pages 82–89. IEEE, 2004. ISBN 0-7695-2112-6. doi:10.1109/HAPTIC.2004.1287181. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1287181. → pages 8, 40

[116] L. A. Jones and D. A. Held. Characterization of Tactors Used in Vibrotactile Displays. *Journal of Computing and Information Science in Engineering*, 8(4):044501, 2008. ISSN 15309827. doi:10.1115/1.2988384. URL http://computingengineering.asmedigitalcollection.asme.org/article.aspx?articleid=1401367. → pages 107

[117] L. A. Jones and I. W. Hunter. A perceptual analysis of stiffness. *Experimental brain research*, 79(1):150, 1990. → pages 120

[118] L. A. L. Jones and M. Berris. The psychophysics of temperature perception and thermal-interface design. In *Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. HAPTICS 2002*, pages 137–142. IEEE Comput. Soc, mar 2002. ISBN 0-7695-1489-8. doi:10.1109/HAPTIC.2002.998951. URL http://dl.acm.org/citation.cfm?id=795682.797526http:

//ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=998951. → pages 9, 136

[119] E. R. Kandel, J. H. Schwartz, and T. M. Jessell. *Principles of Neural Science*. McGraw-Hill, 4th edition, 2000. → pages 2, 9, 135, 141

[120] I. Karuei and K. E. MacLean. Susceptibility to periodic vibrotactile guidance of human cadence. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 141–146. IEEE, feb 2014. ISBN 978-1-4799-3131-6. doi:10.1109/HAPTICS.2014.6775446. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6775446. → pages 14

[121] I. Karuei, K. E. MacLean, Z. Foley-Fisher, R. MacKenzie, S. Koch, and M. El-Zohairy. Detecting vibrations across the body in mobile contexts. In *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*, page 3267, New York, New York, USA, 2011. ACM Press. ISBN 9781450302289. doi:10.1145/1978942.1979426. URL http://dl.acm.org/citation.cfm?doid=1978942.1979426. → pages 2, 10

[122] M. Kim, S. Lee, and S. Choi. Saliency-driven real-time video-to-tactile translation. *IEEE transactions on haptics*, 7(3):394–404, jan 2014. ISSN 2329-4051. doi:10.1109/TOH.2013.58. URL http://www.ncbi.nlm.nih.gov/pubmed/25248221. → pages 18, 58

[123] Y. Kim, J. Cha, I. Oakley, and J. Ryu. Exploring Tactile Movies: An Initial Tactile Glove Design and Concept Evaluation. *IEEE Multimedia*, PP(99):1, 2009. ISSN 1070-986X. doi:10.1109/MMUL.2009.63. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5255212. → pages 8, 13, 17, 18, 19, 40, 41, 42, 43, 44, 84, 136, 138

[124] K. Kuchenbecker, J. Fiene, and G. Niemeyer. Improving contact realism through event-based haptic feedback. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):219–230, mar 2006. ISSN 1077-2626. doi:10.1109/TVCG.2006.32. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1580456. → pages 9

[125] C. Kulkarni, S. P. Dow, and S. R. Klemmer. Early and Repeated Exposure to Examples Improves Creative Work. In *Design Thinking Research*, pages 49–62. 2012. ISBN 978-3-642-31990-7, 978-3-642-31991-4. doi:10.1007/978-3-642-21643-5. URL

http://link.springer.com/chapter/10.1007/978-3-642-31991-4{_}1$\delimiter"026E30F$nhttp://link.springer.com.ezproxy.is.ed.ac.uk/chapter/10.1007/978-3-642-31991-4{_}1$\delimiter"026E30F$nhttp://link.springer.com/content/pdf/10.1007/978-3-642-31991-4{_}1.pdf. → pages 21, 176

[126] S. J. Lederman and R. L. Klatzky. Hand movements: A window into haptic object recognition. *Cognitive Psychology*, 19(3):342–368, jul 1987. ISSN 00100285. doi:10.1016/0010-0285(87)90008-9. URL http://www.sciencedirect.com/science/article/pii/0010028587900089. → pages 7, 11, 135, 141

[127] S. J. Lederman and R. L. Klatzky. Haptic perception: A tutorial. *Attention, Perception & Psychophysics*, 71(7):1439–1459, oct 2009. ISSN 1943-3921. doi:10.3758/APP.71.7.1439. URL http://www.springerlink.com/index/10.3758/APP.71.7.1439. → pages 2, 7, 135

[128] D. Ledo, M. A. Nacenta, N. Marquardt, S. Boring, and S. Greenberg. The HapticTouch toolkit. In *TEI '12*, pages 115–122, New York, USA, Feb. 2012. ACM Press. ISBN 9781450311748. doi:10.1145/2148131.2148157. URL http://dl.acm.org/citation.cfm?id=2148131.2148157. → pages 18, 24, 26, 137

[129] B. Lee, S. Srivastava, R. Kumar, R. Brafman, and S. R. Klemmer. Designing with interactive example galleries. In *CHI '10*, pages 2257–2266, New York, New York, USA, Apr. 2010. ACM Press. ISBN 9781605589299. doi:10.1145/1753326.1753667. URL http://dl.acm.org/citation.cfm?id=1753326.1753667. → pages 21, 62, 64, 65, 74

[130] J. Lee and S. Choi. Evaluation of vibrotactile pattern design using vibrotactile score. In *HAPTICS '12*, pages 231–238. IEEE, Mar. 2012. ISBN 978-1-4673-0809-0. doi:10.1109/HAPTIC.2012.6183796. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6183796. → pages 17, 19, 26, 36, 43, 44, 137

[131] J. Lee and S. Choi. Real-time perception-level translation from audio signals to vibrotactile effects. In *CHI '13*, pages 2567–2576, Paris, France, Apr. 2013. ISBN 9781450318990. doi:10.1145/2470654.2481354. URL http://dl.acm.org/citation.cfm?id=2470654.2481354. → pages 18, 43, 103

[132] J. Lee, J. Ryu, and S. Choi. Vibrotactile score: A score metaphor for designing vibrotactile patterns. In *World Haptics '09*, pages 302–307. IEEE, Mar. 2009. ISBN 978-1-4244-3858-7. doi:10.1109/WHC.2009.4810816. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4810816. → pages 17, 19, 24, 26, 36, 137

[133] J. Lee, Y. Kim, and G. Kim. Funneling and saltation effects for tactile interaction with virtual objects. In *CHI '12*, page 3141, New York, New York, USA, May 2012. ACM Press. ISBN 9781450310154. doi:10.1145/2207676.2208729. URL http://dl.acm.org/citation.cfm?id=2207676.2208729. → pages 10, 41, 46

[134] V. Levesque, L. Oram, K. MacLean, A. Cockburn, N. D. Marchuk, D. Johnson, J. E. Colgate, and M. A. Peshkin. Enhancing physicality in touch interaction with programmable friction. In *Proceedings of the 2011 annual conference on Human factors in computing systems (CHI '11)*, pages 2481–2490, New York, USA, May 2011. ACM Press. ISBN 9781450302289. doi:10.1145/1978942.1979306. URL http://dl.acm.org/citation.cfm?id=1978942.1979306. → pages 8, 130, 131, 136

[135] D. J. Levitin, K. MacLean, M. Mathews, L. Chu, and E. Jensen. The perception of cross-modal simultaneity. *International Journal of Computing and Anticipatory Systems*, 5:323–329, 2000. → pages 2, 144

[136] R. W. Lindeman, J. L. Sibert, E. Mendez-Mendez, S. Patil, and D. Phifer. Effectiveness of directional vibrotactile cuing on a building-clearing task. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '05*, page 271, New York, New York, USA, 2005. ACM Press. ISBN 1581139985. doi:10.1145/1054972.1055010. URL http://portal.acm.org/citation.cfm?doid=1054972.1055010. → pages 14

[137] P. Lopes, A. Ion, and P. Baudisch. Impacto: Simulating physical impact by combining tactile stimulation with electrical muscle stimulation. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15*, pages 11–19, New York, New York, USA, 2015. ACM Press. ISBN 9781450337793. doi:10.1145/2807442.2807443. URL http://dl.acm.org/citation.cfm?doid=2807442.2807443. → pages 12

[138] J. Lofvenberg and R. Johansson. Regional differences and interindividual variability in sensitivity to vibration in the glabrous skin of the human hand. *Brain Research*, 301(1):65–72, 1984. ISSN 00068993. doi:10.1016/0006-8993(84)90403-7. → pages 2, 131

[139] J. Luk, J. Pasquero, S. Little, K. MacLean, V. Levesque, and V. Hayward. A role for haptics in mobile interaction: initial design using a handheld tactile display prototype. In *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06*, page 171, New York, New York, USA, apr 2006. ACM Press. ISBN 1595933727. doi:10.1145/1124772.1124800. URL http://dl.acm.org/citation.cfm?id=1124772.1124800. → pages 9

[140] K. Maclean and M. Enriquez. Perceptual design of haptic icons. In *Eurohaptics*, pages 351–363, 2003. URL http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.138.6172. → pages 14, 19, 63, 64, 108, 110, 166

[141] K. MacLean and V. Hayward. Do It Yourself Haptics: Part II [Tutorial]. *IEEE Robotics & Automation Magazine*, 15(1):104–119, Mar. 2008. ISSN 1070-9932. doi:10.1109/M-RA.2007.914919. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4476335. → pages 19

[142] K. E. MacLean. The 'Haptic Camera': a Technique for Characterising and Playing Back Haptic Properties of Real Environments. In *HAPTICS '96*, Atlanta, GA, 1996. → pages 176

[143] K. E. MacLean. Putting Haptics into the Ambience. *IEEE Transactions on Haptics*, 2(3):123–135, July 2009. ISSN 1939-1412. doi:10.1109/TOH.2009.33. URL http://www.computer.org/portal/web/csdl/doi?doc=doi/10.1109/TOH.2009.33http://portal.acm.org/citation.cfm?id=1608573.1608735. → pages 19, 130

[144] K. E. MacLean, M. J. Enriquez, and T. Lim. Morphing in periodic tactile signals. In *World Haptics 2009*, pages 178–183. IEEE, 2009. ISBN 978-1-4244-3858-7. doi:10.1109/WHC.2009.4810844. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4810844. → pages 9

[145] J. Maloney, M. Resnick, N. Rusk, B. Silverman, and E. Eastmond. The scratch programming language and environment. *Trans. Comput. Educ.*, 10 (4):1–15, 2010. → pages 115

[146] J. Marks, W. Ruml, K. Ryall, J. Seims, S. Shieber, B. Andalman, P. A. Beardsley, W. Freeman, S. Gibson, J. Hodgins, T. Kang, B. Mirtich, and H. Pfister. Design galleries. In *SIGGRAPH '97*, pages 389–400, New York, New York, USA, Aug. 1997. ACM Press. ISBN 0897918967. doi:10.1145/258734.258887. URL http://dl.acm.org/citation.cfm?id=258734.258887. → pages 62, 65

[147] T. Martin and D. L. Schwartz. Physically distributed learning: Adapting and reinterpreting physical environments in the development of fraction concepts. *Cognitive science*, 29(4):587–625, 2005. → pages 115

[148] M. O. Martinez, Minaker Gordon, R. L. Davis, O. Schneider, T. K. Morimoto, A. T. Taylor, A. C. Barron, K. MacLean, P. Blikstein, and A. M. Okamura. HandsOn with Hapkit 3.0: a creative STEM e-learning framework. In *HAPTICS '16 Demos*, 2016. → pages vi

[149] T. H. Massie and J. K. Salisbury. The phantom haptic interface: A device for probing virtual objects. In *Proceedings of the ASME winter annual meeting, symposium on haptic interfaces for virtual environment and teleoperator systems*, volume 55, pages 295–300. Chicago, IL, 1994. → pages 9

[150] S. McCloud. Understanding comics: The invisible art. *Northampton, Mass*, 1993. → pages 180

[151] D. J. Meyer, M. A. Peshkin, and J. E. Colgate. Fingertip friction modulation due to electrostatic attraction. In *2013 World Haptics Conference (WHC)*, pages 43–48. IEEE, apr 2013. ISBN 978-1-4799-0088-6. doi:10.1109/WHC.2013.6548382. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6548382. → pages 8

[152] G. Minaker, O. Schneider, R. Davis, and K. E. MacLean. *HandsOn: Enabling Embodied, Creative STEM e-learning with Programming-Free Force Feedback*, pages 427–437. Springer International Publishing, Cham, 2016. ISBN 978-3-319-42324-1. doi:10.1007/978-3-319-42324-1_42. URL http://dx.doi.org/10.1007/978-3-319-42324-1{_}42. → pages vi, 101

[153] S. O. Modhrain and I. Oakley. Touch TV : Adding Feeling to Broadcast Media. 2001. → pages 13, 19, 84, 165

[154] M. Montessori. *The Advanced Montessori Method*. 1917. → pages 14, 115

[155] C. Moussette. Sketching in Hardware and Building Interaction Design: tools, toolkits and an attitude for Interaction Designers. In *Proc. of Design Research Society*, 2010. URL https://public.me.com/intuitive. → pages 16, 19, 134, 136

[156] C. Moussette and R. Banks. Designing through making. In *TEI '11*, pages 279–282, New York, USA, Jan. 2011. ACM Press. ISBN 9781450304788. doi:10.1145/1935701.1935763. URL http://dl.acm.org/citation.cfm?id=1935701.1935763. → pages 19, 27, 36, 42, 43, 136

[157] C. Moustakas. *Phenomenological Research Methods*. SAGE Publications, Inc., 1994. ISBN 978-0-8039-5798-5. → pages 20, 21, 27, 31, 54, 68, 138

[158] S. Mueller, T. Mohr, K. Guenther, J. Frohnhofen, and P. Baudisch. faBrickation - Fast 3D Printing of Functional Objects by Integrating Construction Kit Building Blocks. In *CHI '14*, pages 3827–3834. ACM Press, Apr. 2014. ISBN 9781450324731. doi:10.1145/2556288.2557005. URL http://dl.acm.org/citation.cfm?id=2611222.2557005. → pages 18

[159] T. Munzner. *Visualization Analysis and Design*. CRC Press, 2014. → pages 175

[160] Myongchan Kim, S. Lee, and Seungmoon Choi. Saliency-Driven Tactile Effect Authoring for Real-Time Visuotactile Feedback. *Haptics: Perception, Devices, Mobility, and Communication*, 7282:258–269, 2012. doi:10.1007/978-3-642-31401-8. URL http://www.springerlink.com/index/10.1007/978-3-642-31401-8. → pages 103

[161] D. A. Norman. *Emotional Design: Why We Love (or Hate) Everyday Things*. Basic Books, New York, NY, USA, 2004. → pages 130

[162] M. Obrist, S. A. Seah, and S. Subramanian. Talking about tactile experiences. In *CHI '13*, pages 1659–1668. ACM Press, 2013. ISBN 9781450318990. doi:10.1145/2470654.2466220. URL http://dl.acm.org/citation.cfm?doid=2470654.2466220. → pages 9, 19, 20, 24, 27, 31, 36, 64, 80, 82, 83, 137

[163] M. Obrist, S. Subramanian, E. Gatti, B. Long, and T. Carter. Emotions Mediated Through Mid-Air Haptics. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*, pages 2053–2062, New York, New York, USA, apr 2015. ACM Press. ISBN 9781450331456. doi:10.1145/2702123.2702361. URL http://dl.acm.org/citation.cfm?id=2702123.2702361. → pages 12, 64

[164] S. Okamoto, H. Nagano, and Y. Yamada. Psychophysical Dimensions of Tactile Perception of Textures. *IEEE Transactions on Haptics*, 6(1):81–93, Jan. 2013. ISSN 1939-1412. doi:10.1109/TOH.2012.32. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6216375. → pages 19, 27, 80, 137

[165] A. Okamura, J. Dennerlein, and R. Howe. Vibration feedback models for virtual environments. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, volume 1, pages 674–679. IEEE, 1998. ISBN 0-7803-4300-X. doi:10.1109/ROBOT.1998.677050. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=677050. → pages 7

[166] A. M. Okamura. Haptic feedback in robot-assisted minimally invasive surgery. *Current Opinion in Urology*, 19(1):102–107, 2009. URL http://www.ncbi.nlm.nih.gov/pubmed/19057225. → pages 9, 12

[167] A. M. Okamura, S. Chan, B. Hannaford, K. MacLean, and W. Provancher. Best Practices for Teaching Haptics, 2012. → pages 19, 136

[168] M. {Orta Martinez}, T. K. Morimoto, A. T. Taylor, A. C. Barron, J. D. A. Pultorak, J. Wang, A. Calasanz-Kaiser, R. L. Davis, P. Blikstein, and A. M. Okamura. 3-D printed haptic devices for educational applications. In *2016 IEEE Haptics Symposium (HAPTICS)*, pages 126–133. IEEE, apr 2016. ISBN 978-1-5090-0903-9. doi:10.1109/HAPTICS.2016.7463166. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=7463166. → pages xx, 9, 15, 115, 116, 119, 136, 182

[169] S. A. Paneels, M. Anastassova, and L. Brunet. TactiPEd: Easy Prototyping of Tactile Patterns. *INTERACT '13*, 8118:228–245, 2013. doi:10.1007/978-3-642-40480-1. URL http://link.springer.com/10.1007/978-3-642-40480-1. → pages 8, 17, 18, 26, 40, 41, 43, 44, 138

[170] S. Papert. *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, 2nd edition, 1980. ISBN 0465046274. URL http://books.google.com/books?id=HhIEAgUfGHwC&pgis=1. → pages 11, 15, 179, 187

[171] J. Park, J. Kim, Y. Oh, and H. Z. Tan. Rendering Moving Tactile Stroke on the Palm Using a Sparse 2D Array. pages 47–56. Springer International Publishing, 2016. doi:10.1007/978-3-319-42321-0_5. URL http://link.springer.com/10.1007/978-3-319-42321-0{_}5. → pages 8

[172] E. W. Pedersen, S. Subramanian, and K. Hornbæk. Is my phone alive? A large-scale study of shape change in handheld devices using videos. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 2579–2588, New York, New York, USA, apr 2014. ACM Press. ISBN 9781450324731. doi:10.1145/2556288.2557018. URL http://dl.acm.org/citation.cfm?id=2611205.2557018. → pages 181

[173] M. Pielot, N. Henze, and S. Boll. Supporting map-based wayfinding with tactile cues. *Proceedings of the 11th International Conference on HumanComputer Interaction with Mobile Devices and Services MobileHCI 09*, page 1, 2009. doi:10.1145/1613858.1613888. URL http://portal.acm.org/citation.cfm?doid=1613858.1613888. → pages 8, 11, 14

[174] M. Prasad, P. Taele, A. Olubeko, and T. Hammond. HaptiGo: A navigational tap on the shoulder'. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 339–345. IEEE, feb 2014. ISBN 978-1-4799-3131-6. doi:10.1109/HAPTICS.2014.6775478. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6775478. → pages 8, 14

[175] D. Prescher, G. Weber, and M. Spindler. A tactile windowing system for blind users. In *Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility*, 12th International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS'10, pages 91–98, New York, New York, USA, 2010. ACM. ISBN 9781605588810. doi:10.1145/1878803.1878821. URL http://www.scopus.com/inward/record.url?eid=2-s2.0-78650615361{&}partnerID=40{&}md5=03aab42397cfae7514aa54752490f023http://portal.acm.org/citation.cfm?doid=1878803.1878821. → pages 9, 11

[176] A. Pusch and A. Lécuyer. Pseudo-haptics: From the Theoretical Foundations to Practical System Design Guidelines. In *Proceedings of the 13th international conference on multimodal interfaces - ICMI '11*, page 57, New York, USA, nov 2011. ACM Press. ISBN 9781450306416. doi:10.1145/2070481.2070494. URL http://dl.acm.org/citation.cfm?id=2070481.2070494. → pages 132

[177] C. Ramstein and V. Hayward. The pantograph. In *Conference companion on Human factors in computing systems - CHI '94*, pages 57–58, New York, New York, USA, 1994. ACM Press. ISBN 0897916514. doi:10.1145/259963.260039. URL http://portal.acm.org/citation.cfm?doid=259963.260039. → pages 9

[178] M. Resnick, B. Myers, K. Nakakoji, B. Shneiderman, R. Pausch, T. Selker, and M. Eisenberg. Design principles for tools to support creative thinking. In *NSF Workshop Report on Creativity Support Tools.*, Washington, DC, 2008. URL http://medcontent.metapress.com/index/A65RM03P4874243N.pdf. → pages 15, 16, 43, 134, 183

[179] P. Richer. A phenomenological analysis of the perception of geometric illusions. *Journal of Phenomenological Psychology*, 8(2):123–135, 1978. → pages 20, 31

[180] J. Rovan and R. Hayward. Typology of Tactile Sounds and their Synthesis in Gesture-Driven Computer Music Performance. *Trends in Gestural Control of Music*, pages 297–320, 2000. → pages 29

[181] J. a. Russell, M. Lewicka, and T. Niit. A cross-cultural study of a circumplex model of affect. *Journal of Personality and Social Psychology*, 57(5):848–856, 1989. ISSN 0022-3514. doi:10.1037//0022-3514.57.5.848. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/0022-3514.57.5.848. → pages 12

[182] J. a. Russell, A. Weiss, and G. a. Mendelsohn. Affect Grid: A single-item scale of pleasure and arousal. *Journal of Personality and Social Psychology*, 57(3):493–502, 1989. ISSN 0022-3514. doi:10.1037//0022-3514.57.3.493. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/0022-3514.57.3.493. → pages 12

[183] G. W. Ryan and H. R. Bernard. Techniques to Identify Themes. *Field Methods*, 15(1):85–109, feb 2003. ISSN 00000000.

doi:10.1177/1525822X02239569. URL
http://fmx.sagepub.com/cgi/doi/10.1177/1525822X02239569. → pages 21, 138

[184] J. Ryu and S. Choi. posVibEditor: Graphical authoring tool of vibrotactile patterns. In *HAVE '08*, pages 120–125. IEEE, Oct. 2008. ISBN 978-1-4244-2668-3. doi:10.1109/HAVE.2008.4685310. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4685310. → pages 17, 42, 43, 64, 65, 137

[185] M. Sato, X. Liu, J. Murayama, K. Akahane, and M. Isshiki. A Haptic Virtual Environment for Molecular Chemistry Education. In *LNCS 5080: Transactions on Edutainment I*, pages 28–39. Springer Berlin Heidelberg, 2008. doi:10.1007/978-3-540-69744-2_3. URL http://link.springer.com/10.1007/978-3-540-69744-2{_}3. → pages 15, 130

[186] M. Sato, X. Liu, J. Murayama, K. Akahane, and M. Isshiki. A haptic virtual environment for molecular chemistry education. In *Transactions on edutainment I*, pages 28–39. Springer, 2008. → pages 115

[187] O. Schneider and K. MacLean. Macaron: An Online, Open-Source, Haptic Editor. In *HAPTICS '16 Demos (finalist for best demo)*, 2016. → pages iv

[188] O. Schneider, S. Zhao, and A. Israr. FeelCraft: User-Crafted Tactile Content. In *Proceedings of AsiaHaptics 2014*, 2014. → pages v, 44, 62, 64, 101, 130, 137, 180

[189] O. Schneider, A. Israr, and K. MacLean. Tactile Animation by Direct Manipulation of Grid Displays. In *UIST '15 Demos*, 2015. → pages iv

[190] O. Schneider, S. Zhao, and A. Israr. Feel Messenger: Embedded Haptics for Social Networking. In *World Haptics '15 Demos*, 2015. → pages v, 101

[191] O. S. Schneider and K. E. MacLean. Improvising Design with a Haptic Instrument. In *HAPTICS '14*, Houston, USA, 2014. → pages iii, 23, 42, 43, 63, 80, 104, 131, 137

[192] O. S. Schneider and K. E. MacLean. Reflections on a WYFIWIF Design Tool. In *Workshop on Tactile User Experience Evaluation Methods at CHI 2014*, 2014. → pages iii

[193] O. S. Schneider and K. E. MacLean. mHIVE: A WYFIWIF design tool. In *2014 IEEE Haptics Symposium (HAPTICS)*, pages 1–1. IEEE, feb 2014. ISBN 978-1-4799-3131-6. doi:10.1109/HAPTICS.2014.6775541. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6775541. → pages iii

[194] O. S. Schneider and K. E. MacLean. Studying Design Process and Example Use with Macaron, a Web-based Vibrotactile Effect Editor. In *HAPTICS '16: Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2016. → pages iv, 61, 137, 138

[195] O. S. Schneider, K. E. MacLean, K. Altun, I. Karuei, and M. M. Wu. Real-time gait classification for persuasive smartphone apps. In *Proceedings of the 2013 international conference on Intelligent user interfaces - IUI '13*, page 161, New York, New York, USA, mar 2013. ACM Press. ISBN 9781450319652. doi:10.1145/2449396.2449418. URL http://dl.acm.org/citation.cfm?id=2449396.2449418. → pages 178

[196] O. S. Schneider, A. Israr, and K. E. MacLean. Tactile Animation by Direct Manipulation of Grid Displays. In *UIST'15*, 2015. → pages iv, 39, 64, 65, 75, 136, 138, 164, 166

[197] O. S. Schneider, H. Seifi, S. Kashani, M. Chun, and K. E. MacLean. HapTurk: Crowdsourcing Affective Ratings for Vibrotactile Icons. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 3248–3260, New York, New York, USA, may 2016. ACM Press. ISBN 9781450333627. doi:10.1145/2858036.2858279. URL http://dl.acm.org/citation.cfm?id=2858036.2858279. → pages iv, 79, 164

[198] D. A. Schön. *The Reflective Practitioner*. 1982. → pages 15, 16, 21, 64, 134, 162, 173, 175, 187

[199] Y. S. Sefidgar, K. E. MacLean, S. Yohanan, H. M. Van der Loos, E. A. Croft, and E. J. Garland. Design and Evaluation of a Touch-Centered Calming Interaction with a Social Robot. *IEEE Transactions on Affective Computing*, 7(2):108–121, apr 2016. ISSN 1949-3045. doi:10.1109/TAFFC.2015.2457893. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7161320. → pages 13, 125

[200] H. Seifi, C. Anthonypillai, and K. E. MacLean. End-user customization of affective tactile messages: A qualitative examination of tool parameters. In

HAPTICS '14, pages 251–256. IEEE, Feb. 2014. ISBN 978-1-4799-3131-6. doi:10.1109/HAPTICS.2014.6775463. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6775463. → pages 2, 19, 42, 62, 64, 80, 102, 137, 180

[201] H. Seifi, K. Zhang, and K. MacLean. VibViz: Organizing, visualizing and navigating vibration libraries. In *World Haptics '15*, 2015. → pages 2, 19, 62, 64, 66, 67, 74, 80, 82, 83, 84, 85, 86, 98, 126, 131, 137

[202] J. Seo and S. Choi. Initial study for creating linearly moving vibrotactile sensation on mobile device. In *HAPTICS '10*, pages 67–70. IEEE, Mar. 2010. ISBN 978-1-4244-6821-8. doi:10.1109/HAPTIC.2010.5444677. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5444677. → pages 10

[203] J. Seo and S. Choi. Perceptual analysis of vibrotactile flows on a mobile device. *IEEE transactions on haptics*, 6(4):522–7, jan 2013. ISSN 2329-4051. doi:10.1109/TOH.2013.24. URL http://www.ncbi.nlm.nih.gov/pubmed/24808404http://www.computer.org/csdl/trans/th/2013/04/tth2013040522.html. → pages 40, 41, 46, 50, 57, 64, 82

[204] M. J. Shaver. *The Twiddler: A Haptic Teaching Tool: Low-Cost Communication and Mechanical Design*. PhD thesis, UBC, 2003. → pages 9

[205] B. Shneiderman. Creating creativity: user interfaces for supporting innovation. *ACM Transactions on Computer-Human Interaction*, 7(1): 114–138, Mar. 2000. ISSN 10730516. doi:10.1145/344949.345077. URL http://dl.acm.org/citation.cfm?id=344949.345077. → pages 15, 16, 134

[206] B. Shneiderman. Creativity support tools: accelerating discovery and innovation. *Communications of the ACM*, 50(12):20–32, dec 2007. ISSN 00010782. doi:10.1145/1323688.1323689. URL http://dl.acm.org/ft{_}gateway.cfm?id=1323689{&}type=html. → pages 176, 183

[207] C. D. Shultz, M. A. Peshkin, and J. E. Colgate. Surface haptics via electroadhesion: Expanding electrovibration with Johnsen and Rahbek. In *2015 IEEE World Haptics Conference (WHC)*, pages 57–62. IEEE, jun 2015. ISBN 978-1-4799-6624-0. doi:10.1109/WHC.2015.7177691. URL

http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7177691. → pages 8

[208] A. Singhal and L. A. Jones. Space-time interactions and the perceived location of cold stimuli. In *2016 IEEE Haptics Symposium (HAPTICS)*, pages 92–97. IEEE, apr 2016. ISBN 978-1-5090-0903-9. doi:10.1109/HAPTICS.2016.7463161. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7463161. → pages 10

[209] J. Smith and K. MacLean. Communicating emotion through a haptic link: Design space and methodology. *International Journal of Human-Computer Studies*, 65(4):376–387, apr 2007. ISSN 10715819. doi:10.1016/j.ijhcs.2006.11.006. URL http://linkinghub.elsevier.com/retrieve/pii/S1071581906001911. → pages 13

[210] R. Sodhi, I. Poupyrev, M. Glisson, and A. Israr. AIREAL. *ACM Transactions on Graphics*, 32(4):1, July 2013. ISSN 07300301. doi:10.1145/2461912.2462007. URL http://dl.acm.org/citation.cfm?id=2461912.2462007. → pages 9, 40

[211] C. Spence and A. Gallace. Multisensory design: Reaching out to touch the consumer. *Psychology and Marketing*, 28(3):267–308, mar 2011. ISSN 07426046. doi:10.1002/mar.20392. URL http://doi.wiley.com/10.1002/mar.20392. → pages 11

[212] J. C. Stevens. Aging and Spatial Acuity of Touch. *Journal of Gerontology*, 47(1):P35–P40, jan 1992. ISSN 0022-1422. doi:10.1093/geronj/47.1.P35. URL http://geronj.oxfordjournals.org/cgi/doi/10.1093/geronj/47.1.P35. → pages 2, 131

[213] J. C. Stevens and K. K. Choo. Spatial Acuity of the Body Surface over the Life Span. *Somatosensory & Motor Research*, 13(2):153–166, 1996. doi:10.3109/08990229609051403. → pages 2, 131

[214] S. Swaminathan, T. Roumen, R. Kovacs, D. Stangl, S. Mueller, and P. Baudisch. Linespace: A sensemaking platform for the blind. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 2175–2185, New York, New York, USA, may 2016. ACM Press. ISBN 9781450333627. doi:10.1145/2858036.2858245. URL http://dl.acm.org/citation.cfm?id=2858036.2858245. → pages 188

[215] C. Swindells, E. Maksakov, K. MacLean, and V. Chung. The Role of Prototyping Tools for Haptic Behavior Design. In *HAPTICS '06*, pages 161–168. IEEE, 2006. ISBN 1-4244-0226-3. doi:10.1109/HAPTIC.2006.1627084. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1627084. → pages 17, 26, 42, 43, 64, 65, 75, 137

[216] C. Swindells, S. Pietarinen, and A. Viitanen. Medium fidelity rapid prototyping of vibrotactile haptic, audio and video effects. In *HAPTICS '14*, pages 515–521. IEEE, Feb. 2014. ISBN 978-1-4799-3131-6. doi:10.1109/HAPTICS.2014.6775509. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6775509http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6775509. → pages 18, 40, 42, 43, 64, 75, 137, 138

[217] H. Tan, R. Gray, J. J. Young, and R. Taylor. A Haptic Back Display for Attentional and Directional Cueing. 2003. ISSN 1545-1143. → pages 14

[218] H. Tan, A. Lim, and R. Traylor. A psychophysical study of sensory saltation with an open response paradigm. *nternational Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 69: 1109–1115, 2009. → pages 10, 41

[219] H. Z. Tan and A. Pentland. Tactual displays for wearable computing. *Personal Technologies*, 1(4):225–230, dec 1997. ISSN 0949-2054. doi:10.1007/BF01682025. URL http://link.springer.com/10.1007/BF01682025. → pages 14, 110

[220] K. Tanie, S. Tachi, K. Komoriya, M. Abe, K. Asaba, and Y. Tomita. Information Transmission Characteristics of Two-Dimensional Electrocutaneous Phantom Sensation. *Transactions of the Society of Instrument and Control Engineers*, 16(5):732–739, 1980. → pages 10, 50

[221] D. Ternes and K. E. MacLean. Designing Large Sets of Haptic Icons with Rhythm. *Haptics: Perception, Devices and Scenarios*, 5024:199–208, 2008. URL http://www.springerlink.com/index/atl5686222k242m4.pdf. → pages 14, 19, 26, 63, 80, 82, 88

[222] C. Torres, T. Campbell, N. Kumar, and E. Paulos. HapticPrint: Designing Feel Aesthetics for Digital Fabrication. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pages 583–591, New York, NY, USA, 2015. ACM. ISBN

978-1-4503-3779-3. doi:10.1145/2807442.2807492. URL http://doi.acm.org/10.1145/2807442.2807492. → pages 176

[223] D. Tsetserukou, A. Neviarouskaya, H. Prendinger, N. Kawakami, and S. Tachi. Affective haptics in emotional communication. In *Proc. ACII '09*, pages 1–6, Sept. 2009. doi:10.1109/ACII.2009.5349516. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5349516. → pages 17, 41

[224] O. A. J. 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*, 23(6):1180–1190, 2009. URL http://www.ncbi.nlm.nih.gov/pubmed/19118414. → pages 9, 12

[225] C. Vaucelle, L. Bonanni, and H. Ishii. Design of haptic interfaces for therapy. In *CHI '09*, page 467, New York, New York, USA, 2009. ACM Press. ISBN 9781605582467. doi:10.1145/1518701.1518776. URL http://dl.acm.org/citation.cfm?doid=1518701.1518776. → pages 13, 130

[226] R. T. Verrillo and G. A. Gescheider. Perception via the sense of touch. *Tactile aids for the hearing impaired*, pages 1–36, 1992. → pages 50

[227] A. Warr and E. O'Neill. Understanding design as a social creative process. In *Creativity & cognition - C&C '05*, pages 118–127. ACM Press, Apr. 2005. ISBN 1595930256. doi:10.1145/1056224.1056242. URL http://dl.acm.org/citation.cfm?id=1056224.1056242. → pages 15, 16, 62, 64, 69, 74, 134, 173

[228] J. Watanabe, T. Hayakawa, S. Matsui, A. Kano, Y. Shimizu, and M. Sakamoto. Visualization of Tactile Material Relationships Using Sound Symbolic Words. In P. Isokoski and J. Springare, editors, *Haptics: Perception, Devices, Mobility, and Communication*, volume 7283 of *Lecture Notes in Computer Science*, pages 175–180. Springer, Berlin, Heidelberg, 2012. ISBN 978-3-642-31403-2. doi:10.1007/978-3-642-31404-9. URL http://www.springerlink.com/index/10.1007/978-3-642-31404-9. → pages 36, 126

[229] C. E. Wieman, W. K. Adams, and K. K. Perkins. PhET: Simulations that enhance learning. *Science*, 322:682–683, 2008. → pages 115

[230] R. L. Williams, M. Srivastava, R. Conaster, and J. N. Howell. Implementation and evaluation of a haptic playback system. *Haptics-e*, 3: 160–176, 2004. → pages 115

[231] G. Wilson, T. Carter, S. Subramanian, and S. A. Brewster. Perception of ultrasonic haptic feedback on the hand. In *CHI '14*, pages 1133–1142, New York, New York, USA, Apr. 2014. ACM Press. ISBN 9781450324731. doi:10.1145/2556288.2557033. URL http://dl.acm.org/citation.cfm?id=2611105.2557033. → pages 40, 41, 57

[232] L. Winfield, J. Glassmire, J. E. Colgate, and M. Peshkin. T-PaD: Tactile Pattern Display through Variable Friction Reduction. In *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*, pages 421–426. IEEE, mar 2007. ISBN 0-7695-2738-8. doi:10.1109/WHC.2007.105. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4145211. → pages 8, 111

[233] H.-Y. Yao and V. Hayward. Design and analysis of a recoil-type vibrotactile transducer. *The Journal of the Acoustical Society of America*, 128(2):619–627, Aug. 2010. ISSN 1520-8524. doi:10.1121/1.3458852. URL http://link.aip.org/link/?JASMAN/128/619/1. → pages 8, 29, 136, 166

[234] K. Yatani and K. N. Truong. SemFeel: a user interface with semantic tactile feedback for mobile touch-screens. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology - UIST '09*, page 111, New York, New York, USA, 2009. ACM Press. ISBN 9781605587455. doi:10.1145/1622176.1622198. URL http://portal.acm.org/citation.cfm?doid=1622176.1622198. → pages 14

[235] S. Yohanan and K. E. MacLean. A tool to study affective touch. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, CHI EA '09, pages 4153–4158, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-247-4. doi:http://doi.acm.org/10.1145/1520340.1520632. URL http://doi.acm.org/10.1145/1520340.1520632. → pages 13

[236] S. Yohanan and K. E. MacLean. Design and assessment of the haptic creature's affect display. In *Proceedings of the 6th international conference on Human-robot interaction*, HRI '11, pages 473–480, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0561-7.

doi:http://doi.acm.org/10.1145/1957656.1957820. URL http://doi.acm.org/10.1145/1957656.1957820. → pages 10, 172

[237] S. Yohanan and K. E. MacLean. The Role of Affective Touch in Human-Robot Interaction: Human Intent and Expectations in Touching the Haptic Creature. *International Journal of Social Robotics*, 4(2):163–180, dec 2011. ISSN 1875-4791, 1875-4805. doi:10.1007/s12369-011-0126-7. URL http://www.springerlink.com/index/10.1007/s12369-011-0126-7. → pages 10, 12, 172

[238] S. Yohanan, M. Chan, J. Hopkins, H. Sun, and K. MacLean. Hapticat: exploration of affective touch. In *Proceedings of the 7th international conference on Multimodal interfaces*, ICMI '05, pages 222–229, New York, NY, USA, 2005. ACM. ISBN 1-59593-028-0. doi:http://doi.acm.org/10.1145/1088463.1088502. URL http://doi.acm.org/10.1145/1088463.1088502. → pages 13

[239] Yongjae Yoo, Taekbeom Yoo, Jihyun Kong, and Seungmoon Choi. Emotional responses of tactile icons: Effects of amplitude, frequency, duration, and envelope. In *2015 IEEE World Haptics Conference (WHC)*, pages 235–240. IEEE, jun 2015. ISBN 978-1-4799-6624-0. doi:10.1109/WHC.2015.7177719. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7177719. → pages 12, 64

[240] T. Yoo, Y. Yoo, and S. Choi. An Explorative Study on Crossmodal Congruence Between Visual and Tactile Icons Based on Emotional Responses. In *ICMI '14*, pages 96–103. ACM Press, Nov. 2014. ISBN 9781450328852. doi:10.1145/2663204.2663231. URL http://dl.acm.org/citation.cfm?id=2663204.2663231. → pages 8

[241] S. Zhao, O. Schneider, R. Klatzky, J. Lehman, and A. Israr. FeelCraft: Crafting Tactile Experiences for Media using a Feel Effect Library. In *UIST '14 Demos*, 2014. URL http://dl.acm.org/citation.cfm?id=2659109. → pages v, 101

[242] Y. Zheng and J. B. Morrell. Haptic actuator design parameters that influence affect and attention. In *2012 IEEE Haptics Symposium (HAPTICS)*, pages 463–470. IEEE, Mar. 2012. ISBN 978-1-4673-0809-0. doi:10.1109/HAPTIC.2012.6183832. URL http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=6183832. → pages 19, 27, 36, 137

[243] C. Zilles and J. Salisbury. A constraint-based god-object method for haptic display. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, volume 3, pages 146–151. IEEE Comput. Soc. Press, 1995. ISBN 0-8186-7108-4. doi:10.1109/IROS.1995.525876. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=525876. → pages 9