INTERACTIVE SYSTEMS BASED ON ELECTRICAL MUSCLE STIMULATION

Pedro Lopes
University of Potsdam & Hasso Plattner Institute

Advisor: Prof. Dr. Patrick Baudisch
Second advisor: Prof. Dr. Felix Naumann

This dissertation is submitted for the degree of Doctor of Science
– Dr. rer. nat. –
December 2018

This work is licensed under a Creative Commons License:
Attribution – Noncommercial – 4.0 International
To view a copy of this license visit
https://creativecommons.org/licenses/by-nc/4.0/

Published online at the
Institutional Repository of the University of Potsdam:
URN urn:nbn:de:kobv:517-opus4-421165
https://nbn-resolving.org/urn:nbn:de:kobv:517-opus4-421165

This thesis is dedicated to all the kind souls that supported me throughout:
my family (Sofia, Maria, Paulo, João and Maria Graça),
my dearest friends (Ava and Constanze),
my advisor Patrick and his team at HPI,
and the writer David Forster Wallace.

Δ

We shall not cease from exploration
And the end of all our exploring
Will be to arrive where we started
And know the place for the first time.

— T.S. Elliot (*Little Gidding*, in *Four Quartets*, 1943)

DECLARATION

This dissertation is the result of my own work and collaborations in which I was always the scientific lead. This dissertation has not been previously submitted, in part or whole, to any university or institution for any degree, diploma, or other qualification. The ideas herein have appeared in the following published papers:

1. Adding Force Feedback to mixed reality Experiences and Games using Electrical Muscle Stimulation. **Pedro Lopes**, Sijing You, Alexandra Ion, and Patrick Baudisch. *In Proc. CHI'18*. Paper 446.
2. Providing Haptics to Walls & Heavy Objects in Virtual Reality by Means of Electrical Muscle Stimulation. **Pedro Lopes,** Sijing You, Lung-Pan Cheng, Sebastian Marwecki, and Patrick Baudisch. *In Proc. CHI'17*, pg. 1471-1482.
3. Muscle-plotter: An Interactive System based on Electrical Muscle Stimulation that Produces Spatial Output. **Pedro Lopes**, Doga Yüksel, François Guimbretière, and Patrick Baudisch. *In Proc. UIST '16*, pg. 207-217.
4. Impacto: Simulating Physical Impact by Combining Tactile Stimulation with Electrical Muscle Stimulation. **Pedro Lopes**, Alexandra Ion, and Patrick Baudisch. *In Proc. UIST '15*, pg. 11-19.
5. Affordance++: Allowing Objects to Communicate Dynamic Use. **Pedro Lopes**, Patrik Jonell, and Patrick Baudisch. *In Proc. CHI '15*, pg. 2515-2524.
6. Proprioceptive Interaction. **Pedro Lopes**, Alexandra Ion, Willi Mueller, Daniel Hoffmann, Patrik Jonell, and Patrick Baudisch. *In Proc. CHI '15*, pg. 939-948.
7. Muscle-propelled force feedback: bringing force feedback to mobile devices. **Pedro Lopes** and Patrick Baudisch. *In Proc. CHI '13*, pg. 2577-2580.

Signed:

Date: 5th May 2018

Pedro Lopes

ABSTRACT

How can interactive devices connect with users in the most immediate and intimate way? This question has driven interactive computing for decades. Throughout the last decades, we witnessed how mobile devices moved computing into users' pockets, and recently, wearables put computing in constant physical contact with the user's skin. In both cases moving the devices closer to users allowed devices to sense more of the user, and thus act more personal. The main question that drives our research is: what is the next logical step?

Some researchers argue that the next generation of interactive devices will move past the user's skin and be directly implanted inside the user's body. This has already happened in that we have pacemakers, insulin pumps, etc. However, we argue that what we see is not devices moving towards the inside of the user's body, but rather towards the body's biological "interface" they need to address in order to perform their function.

To implement our vision, we created a set of devices that intentionally *borrow* parts of the user's body for input and output, rather than adding more technology to the body.

In this dissertation we present one specific flavor of such devices, i.e., devices that *borrow the user's muscles*. We engineered I/O devices that interact with the user by reading and controlling muscle activity. To achieve the latter, our devices are based on medical-grade signal generators and electrodes attached to the user's skin that send electrical impulses to the user's muscles; these impulses then cause the user's muscles to contract.

While electrical muscle stimulation (EMS) devices have been used to regenerate lost motor functions in rehabilitation medicine since the 1960s, in this dissertation, we propose a new perspective: EMS as a means for creating interactive systems.

We start by presenting seven prototypes of interactive devices that we have created to illustrate several benefits of EMS. These devices form two main categories: (1) Devices that allow users eyes-free access to information by means of their proprioceptive sense, such as the value of a variable in a computer system, a tool, or a plot; (2) Devices that increase immersion in virtual reality by simulating large forces, such as wind, physical impact, or walls and heavy objects.

Then, we analyze the potential of EMS to build interactive systems that miniaturize well and discuss how they leverage our proprioceptive sense as an I/O modality. We proceed by laying out the benefits and disadvantages of both EMS and mechanical haptic devices, such as exoskeletons.

We conclude by sketching an outline for future research on EMS by listing open technical, ethical and philosophical questions that we left unanswered.

ZUSAMMENFASSUNG

Wie können interaktive Geräte auf unmittelbare und eng verknüpfte Weise mit dem Nutzer kommunizieren? Diese Frage beschäftigt die Forschung im Bereich Computer Interaktion seit Jahrzehnten. Besonders in den letzten Jahren haben wir miterlebt, wie Nutzer interaktive Geräte dauerhaft bei sich führen, im Falle von sogenannten Wearables sogar als Teil der Kleidung oder als Accessoires. In beiden Fällen sind die Geräte näher an den Nutzer gerückt, wodurch sie mehr Informationen vom Nutzer sammeln können und daher persönlicher erscheinen. Die Hauptfrage, die unsere Forschung antreibt, ist: Was ist der nächste logische Schritt in der Entwicklung interaktiver Geräte?

Mache Wissenschaftler argumentieren, dass die Haut nicht mehr die Barriere für die nächste Generation von interaktiven Geräten sein wird, sondern dass diese direkt in den Körper der Nutzer implantiert werden. Zum Teil ist dies auch bereits passiert, wie Herzschrittmacher oder Insulinpumpen zeigen. Wir argumentieren jedoch, dass Geräte sich in Zukunft nicht zwingend innerhalb des Körpers befinden müssen, sondern sich an der richtigen „Schnittstelle“ befinden sollen, um die Funktion des Gerätes zu ermöglichen.

Um diese Entwicklung voranzutreiben haben wir Geräte entwickelt, die Teile des Körpers selbst als Ein- und Ausgabe-Schnittstelle *verwenden*, anstatt weitere Geräte an den Körper anzubringen.

In dieser Dissertation zeigen wir eine bestimmte Art dieser Geräte, nämlich solche, die *Muskeln verwenden.* Wir haben Ein-/Ausgabegeräte gebaut, die mit dem Nutzer interagieren indem sie Muskelaktivität erkennen und kontrollieren. Um Muskelaktivität zu kontrollieren benutzen wir Signalgeber von medizinischer Qualität, die mithilfe von auf die Haut geklebten Elektroden elektrische Signale an die Muskeln des Nutzers senden. Diese Signale bewirken dann eine Kontraktion des Muskels.

Geräte zur elektrischen Muskelstimulation (EMS) werden seit den 1960er-Jahren zur Regeneration von motorischen Funktionen verwendet. In dieser Dissertation schlagen wir jedoch einen neuen Ansatz vor: elektrische Muskelstimulation als Kommunikationskanal zwischen Mensch und interaktiven Computersysteme.

Zunächst stellen wir unsere sieben interaktiven Prototypen vor, welche die zahlreichen Vorteile von EMS demonstrieren. Diese Geräte können in zwei Hauptkategorien unterteilt werden: (1) Geräte, die Nutzern Zugang zu Information direkt über ihre propriozeptive Wahrnehmung geben ohne einen visuellen Reiz. Diese Informationen können zum Beispiel Variablen, Diagramme oder die Handhabung von Werkzeugen beinhalten. (2) Des Weiteren zeigen wir Geräte, welche die Immersion in virtuelle Umgebungen erhöhen indem sie physikalische Kräfte wie Wind, physischen Kontakt, Wände oder schwere Objekte, simulieren.

Wir analysieren in dieser Arbeit außerdem das Potential von EMS für miniaturisierte interaktive Systeme und diskutieren, wie solche EMS Systeme die propriozeptive Wahrnehmung wirksam als Ein-/Ausgabemodalität nutzen können. Dazu stellen wir die Vor- und Nachteile von EMS und mechanisch-haptischen Geräten, wie zum Beispiel Exoskeletten, gegenüber.

Zum Abschluss skizzieren wir zukünftige Richtungen in der Erforschung von interaktiven EMS Systemen, indem wir bislang offen gebliebene technische, ethische und philosophische Fragen aufzeigen.

ACKNOWLEDGEMENTS

Foremost, I would like to thank my advisor Patrick Baudisch. In fact, besides being my advisor and mentor, Patrick was always my closest collaborator in all these projects. I owe much to Patrick in his ability to solve many of the practical aspects of this dissertation, but also its deepest philosophical riddles. Ultimately, it was a blessing to have a collaborator with 24–7 enthusiasm & dedication directed at answering some of the hard questions that arose from our investigations, many of them are still unanswered—but isn't that what research is? After these years at HPI, I feel that Patrick taught me most of what I know about research, teaching, problem solving and—I dare to say—about staying true to your research vision and I am very thankful for that opportunity.

I would like to thank my family & friends for their unconditional support as I moved to Berlin to work on this dissertation.

I would like to thank the whole Human Computer Interaction (HCI) community. This work is inspired, derived, remixed by all of your work. I would like to especially thank the CHI and UIST communities for their valuable feedback and inspiring scientific programs. I would also like to thank Prof. Jun Rekimoto (Tokyo University) and Prof. Hiroshi Ishii (MIT Media Lab) who never fell short of inspiring me with their own visionary thoughts and ideas. In the same vein, I would like to thank my fellow colleagues that have taken on the challenge of advancing the field of electrical muscle stimulation and never ceased to inspire: Max Pfeiffer (University of Münster), Jun Nishida (Tsukuba University) and Shunichi Kasahara (Sony CSL).

Dissertations always end up looking as a solo effort once the work is written up. This could not be further from the truth. This work could not have been done without the wonderful team at the Human Computer Interaction lab and their unbelievable patience, support, inspiration and dedication. In particular, amongst the HCI team, a special thanks goes out to Robert Kovacs, Alexandra Ion, David Lindlbauer (TU), Anne Roudaut, Lung-Pan Cheng, Stefanie Müller, François Guimbretière (Cornell) and Sebastian Marwecki; these folks had the patience to spend hours with me co-authoring many of my projects. Thanks also to the wise advice of my senior colleagues Dominik Schmidt, Sean Gustafson, Bruno Araújo, and Oliver Schneider—all of them kept on inspiring me throughout.

A very special thanks goes out to my dearest friend and colleague Ricardo Jota, who—as the incredible wise and centered person he is—pushed all the right buttons in my brain to show me this way. A heartfelt thank-you goes out to Nicolai Marquardt (UCL) for his patience, kindness and guidance. Also, this thank-you note to my sensei would not be complete without thanking Tiago Ribeiro, who sat beside me for three years and taught me how to think programmatically—"nha".

Deeply, I would like to thank the three wonderful master students that I co-advised at HPI: Patrik Jonell, Sijing You, Doga Yueksel, for their endless patience and inspiration. If there is any sense to the denomination "master student" is because they were my masters in many ways.

I would like now to risk drafting a very long list to thank the many brilliant students I had the pleasure to interact with at HPI: Fabian Bornhofen, Anton Gulenko, Lars Wassermann, Lena Herscheid, Alex Teibrich, Marius Knaust, Markus Dreseler, Patrick Lühne, Willi Müller, Daniel Hoffmann, Dustin Beyer, Astrid Thomschke, Andreas Rau, Christoph Sterz, Max Plauth, Martin Fritzsche, Maximilian Schneider, and Lars Butzmann. Thanks for believing in my ideas, even when these did not work out.

Also, venturing again into long lists: thanks to André Gonçalves (ADDAC) and Uwe Hentschel (HPI) who helped in solving some of the technical puzzles in pose-IO; thanks to Philipp Herholz and Désirée Förster for the assistance in translating the abstract; to Ludwig Wall and Thijs Roumen for helping out countless times; thanks to Profs. Joaquim Jorge, Alfredo Ferreira and Tiago Guerreiro for providing me solid ground at IST Lisbon; thanks once more to my father (Prof. Paulo Lopes)—to whom the thesis is also dedicated—for assisting in proofreading; and, thanks to Christoph Peschel, Michael Karsch, and Arthur Silber for making HPI a friendlier workplace.

Furthermore, this research could not have been done outside the generous support of many: thanks to the Hasso Plattner Institute (HPI) for creating superb working conditions throughout and thanks to the HPI Research School scholarship for their support; with a special thank you to Profs. Andreas Polze, Felix Naumann and Robert Hirschfeld for their valuable feedback.

The ideas presented in this thesis were not always inspired by advancements in HCI but, very often, from thinkers and doers that transcend the labeling of fields. I would like to thank the artists Stelarc, Julian Oliver and Daito Manabe for their inspiration. Many a time these ideas were also born outside of research, sometimes, when improvising music. This force of expression, which renews mind, body and soul, has been kept alive by the unconditional support of my dearest friends: Ondness, Pedro Sousa, Gabriel Ferrandini, Kazehito Seki, Paolo Gaiba Riva, Thessia Machado, Bruno Teles, Stefan Brunner and Igor Sousa.

Lastly, as odd as it may sound to the reader, I would like to thank the S7 train, where most of my papers were written. *Next stop, Griebnitzsee, in the direction of Chicago.*

Δ

CONTENTS

1 Introduction 19
- 1.1 An example of an interactive system based on EMS 20
- 1.2 Our contributions 21
- 1.3 Two lines of exploration 21
- 1.4 Structure of this dissertation 22

2 Related Work 23
- 2.1 Haptics 23
  - 2.1.1 Tactile feedback 23
  - 2.1.2 Proprioceptive feedback 25
  - 2.1.3 Motor-based force feedback 26
- 2.2 Wearables based on Proprioceptive Input 27
  - 2.2.1 Wearables with motion tracking (IMUs and computer vision) 27
  - 2.2.2 Wearables that read muscle activity (EMGs and MMGs) 28
  - 2.2.3 Eyes-free interaction 29
- 2.3 Electrical muscle stimulation 30
  - 2.3.1 Working principle 30
  - 2.3.2 EMS signal generator 31
  - 2.3.3 Deploying an EMS system 31
  - 2.3.4 Application areas of electrical muscle stimulation 32

3 Increasing immersion using EMS 36
- 3.1 Muscle-propelled Force Feedback: force feedback for mobile devices 36
  - 3.1.1 Device hardware 37
  - 3.1.2 Contributions, benefits, and limitations 38
  - 3.1.3 Study 1: measuring EMS output force 38
  - 3.1.4 Study 2: mobile force feedback gaming 40
  - 3.1.5 Conclusions on muscle-propelled force feedback 41
- 3.2 Impacto: simulating impacts in virtual reality 42
  - 3.2.1 Impacto’s two components are mutually beneficial 44
  - 3.2.2 Contributions, benefits, and limitations 44
  - 3.2.3 Application examples 45
  - 3.2.4 Implementation details 49
  - 3.2.5 User study 52
  - 3.2.6 Conclusions on impacto 56
- 3.3 Providing haptics to walls & heavy objects 57
  - 3.3.1 Contributions, benefits, and limitations 59
  - 3.3.2 Design 59
  - 3.3.3 User study 1: validating designs 62
  - 3.3.4 Implementation details 66
  - 3.3.5 Example widgets 68
  - 3.3.6 User study 2: the experience 73

3.3.7 Conclusions on EMS for VR walls & heavy objects 75
3.4 Hands-free force feedback for mixed reality 77
3.4.1 Walkthrough of our mixed reality experience 78
3.4.2 Summary of walkthrough 83
3.4.3 Demonstrating EMS haptics in mixed reality experiences 84
3.4.4 Contributions, benefits, and limitations 86
3.4.5 Implementation 86
3.4.6 User study 89
3.4.7 Discussion of our findings 92
3.4.8 Conclusions on EMS as mixed reality force feedback 93
3.5 EMS & mechanical-actuation: sides of the same coin? 94
3.5.1 Similarities between EMS and mechanical actuation 94
3.5.2 Advantages of mechanical actuation 95
3.5.3 Advantages of EMS-based actuation 95

4 Information access using EMS 96
4.1 Proprioceptive Interaction: poses for input and output 97
4.1.1 The pose-IO device 98
4.1.2 Applications 99
4.1.3 Contributions, benefits, and limitations 102
4.1.4 Implementation details 103
4.1.5 Study of proprioceptive input & output 106
4.1.6 An exploratory study of the proprioceptive interfaces 109
4.1.7 Conclusions on proprioceptive interaction 111
4.2 Affordance++ 113
4.2.1 Affordance++: allowing objects to communicate their use 114
4.2.2 Illustrating some use cases of affordance++ 115
4.2.3 Contributions, benefits, and limitations 119
4.2.4 Prototypes 120
4.2.5 User study 122
4.2.6 Study task 1: understanding the meaning of induced motion 123
4.2.7 Study task 2: affordance++ communicating use 126
4.2.8 Conclusions on affordance++ 130
4.2.9 Reflecting on the quality of our information access via EMS 130
4.3 Muscle-Plotter 132
4.3.1 Walkthrough: an interactive wind tunnel simulator 132
4.3.2 More application scenarios 135
4.3.3 Summary of widgets 138
4.3.4 Contributions, benefits, and limitations 138
4.3.5 Implementation details 139
4.3.6 User study: validating our implementation 145
4.3.7 Conclusions on muscle-plotter 148
4.4 Discussion on information access using EMS 150

4.4.1 Implementing proprioceptive interaction with alternative actuation hardware 150
4.4.2 Limited bandwidth 150
4.4.3 Beyond hands and upper limbs 150
4.4.4 Information access vs. learning a new skill 151

5 Conclusions & Discussion 152
5.1 Benefits of interactive systems based on EMS 152
5.1.1 Miniaturization of force feedback 152
5.1.2 Next step after wearables: devices becoming more personal 153
5.1.3 Bi-directional communication channel to the user 154
5.2 Outline and open challenges 154
5.2.1 Technical challenges 154
5.2.2 Conceptual, philosophical & societal challenges 155
5.3 Community outreach 157
5.4 Final thoughts 157

6 References 158

LIST OF FIGURES

Figure 1: Our proposed interactive systems borrow parts of the user's body, making the resulting devices effectively overlap with the user's body.. 19

Figure 2: An interactive system based on electrical muscle stimulation: pose-IO......... 20

Figure 3: Our two lines of work in interactive systems based on EMS. 21

Figure 4: Illustrating the weight-force tradeoff in exoskeletons by comparing (a) the stationary EXO-UL7 and (b) the portable SAM exoskeleton (images from UCLA Bionics Lab press kit and with permission of Université Libre de Bruxelles).............. 27

Figure 5: Electrically actuating the biceps muscle using EMS. 32

Figure 6: Timeline of EMS research in the field of Human Computer Interaction. (To preserve visual clarity we do not display publications from adjunct proceedings or other non-archival formats, except the early work by Farbiz et al.)....................................... 33

Figure 7: Our prototype electrically stimulates the user's arm muscles via the shown electrodes, causing the user to involuntarily tilt the device. As the user is countering this force, they perceive force feedback.. 37

Figure 8: (a) As the user is playing (b) muscle-propelled force feedback kicks in, causing the user's left wrist to tilt the device. (c) The user responds by countering the forces, steering the plane against the wind.. 37

Figure 9: The backside of our prototype reveals its hardware design........................... 38

Figure 10: Apparatus for measuring the force an EMS-induced contraction................ 39

Figure 11: Average peak force (in grams) during palm flexion for each intensity. 39

Figure 12: Participants from Study 2 experiencing the force feedback sensations delivered by our prototype (photos with participants' consent)................................... 41

Figure 13: Impacto is designed to render the sensation of hitting and being hit........... 42

Figure 14: The solenoid component with a "knuckle" tip (c), which has a 90° lever to hit the skin orthogonally. The other four interchangeable tips are (a) generic surface, (b) small generic surface, (d) rounded, and (e) sharp... 43

Figure 15: Detail of the electrical muscle stimulation component. Here, it stimulates the user's *biceps brachii* muscles causing an involuntary contraction that resembles force feedback.. 43

Figure 16: The impacto bracelet opened to reveal its contents. 44

Figure 17: Stereo headset view from our simple boxing simulation. It allows users to attack the avatar and to block the avatar's attacks. Here, we see the avatar attacking the user's right arm.. 45

Figure 18: (a) Impacto allows users to feel the impact of blocking the avatar's hit by thrusting the user's arm backwards by operating the user's biceps. (b) The same impacto unit allows simulating the sensation of attacking. 46

Figure 19: Additional solenoid unit mounted to the back of the hand featuring a surface tip allows the user to attack using jabs and uppercuts. 46

Figure 20: By wearing impacto on the leg and foot, the user experiences the impact of kicking a virtual football. 47

Figure 21: Stereo headset view from the football-juggling simulator. Here, the impacto unit renders the impact of a ball on the user's foot. 47

Figure 22: Close-up of on impacto unit mounted on the user's instep and calf. The wireless accelerometer on the solenoid senses the foot's tilt. 48

Figure 23: A Thai Boxing experience with impacto. 48

Figure 24: User feeling the impact of hitting a virtual baseball. 49

Figure 25: Achieving multiple haptic experiences via the same combination of impacto and a physical prop. 49

Figure 26: Schematic of impacto's circuitry. 50

Figure 27: Apparatus for measuring force and latency. 51

Figure 28: Our experimental setup to test impacto. 53

Figure 29: Realism ratings in dependence of force feedback (*EMS*) and tactile feedback (*solenoid*) conditions. 54

Figure 30: (a) As this user lifts a virtual cube, our system lets the user feel the weight and resistance of the cube. (b) Our system implements this by actuating the user's *opposing* muscles using electrical muscle stimulation. 57

Figure 31: (a) When a user picks up a physical cube, its weight causes tension in the user's biceps. (b) Our system creates this tension by instead actuating the opposing muscles, here the user's triceps and shoulders. 58

Figure 32: We use up to 8 electrode pairs, actuating (a) wrist, (b) biceps, (c) triceps, and (d) shoulders. 58

Figure 33: Implementing rigid walls requires stimulating muscles with strong impulses over long periods. This draws undesired attention to the electrical stimulation. 60

Figure 34: The *magnetic* visuals allow the user's hand to penetrate the objects. 61

Figure 35: The *electro* visuals. 61

Figure 36: Study participant in the virtual world of the study, surrounded by five virtual walls; each wall embodies a study condition. 63

Figure 37: Eight participants picked the *repulsion* wall as their favorite design. 64

Figure 38: Participants rated the *repulsion* wall as the most realistic. 64

Figure 39: Participants rated the EMS conditions more impermeable than *vibro*.......... 65

Figure 40: The *repulsion* wall stopped participants' hands on average 3.6 cm (error bars denote std. dev.).. 65

Figure 41: The battery powered, medically compliant, muscle stimulator we used. 66

Figure 42: Electrode placement for arm and shoulder.. 67

Figure 43: The jail cell features "electrified" walls and a gate. Touching any of these repel the user's hand.. 69

Figure 44: Pushing this soft button opens the gate.. 70

Figure 45: This cannon shoots projectiles that we render using the *repulsion* effect..... 70

Figure 46: The user is dragging the knob of a slider mechanism................................. 71

Figure 47: Two cubes that users can push onto the button on the right or pick up. 71

Figure 48: The user picked up a cube and is about to throw it over the glass............... 72

Figure 49: Users have to drive this cube up the ramp by punching it. The system responds with a mix between a soft effect and a repulsion effect. 72

Figure 50: Participants rated the *EMS* condition higher than in baseline...................... 74

Figure 51: Participants preferred the experience in the *EMS* condition. 74

Figure 52: In this mixed reality game that uses a physical tray as prop, our mobile EMS system renders shifts in the tray's center of gravity as the marble moves. 77

Figure 53: Using a cup, actuated by means of EMS, as a tangible brightness dial. 78

Figure 54: The previous scene through the HoloLens.. 79

Figure 55: (a) This user physically drags the couch and feels the EMS simulated friction against the floor. (b) As the couch collides with a real wall, the system stops the user by pushing the user's shoulders and wrists backwards. 79

Figure 56: Our EMS system renders the forces of the button's mechanism. 80

Figure 57: The user configures the intensity of the desired light bulb using a cup as a stand-in for a dial (passive prop). Using EMS force feedback, our system augments the tangible with constraints and detents... 81

Figure 58: The user manipulates two cups to control the light temperature and intensity simultaneously.. 82

Figure 59: Here, our system enhances a fully functional thermostat with detents......... 83

Figure 60: Walkthrough examples mapped to the reality-virtuality continuum............ 83

Figure 61: While the user pulls the lever of this virtual catapult, our system provides force feedback simulating the catapult's spring. 84

Figure 62: (a) At the start of the game the marble falls from the sky. (b) As it hits the tray, the EMS pulls the user's arms down quickly so as to represent the impact.......... 85

Figure 63: (a) These gooseneck lamps are repurposed as levers, with force feedback, allowing the user to input the secret combination to (b) unlock the door. 85

Figure 64: The hardware components and electrode placement (one arm only)........... 87

Figure 65: Participants rated their experience as more realistic with EMS.................. 90

Figure 66: Participants rated their experience as more enjoyable in most of the EMS conditions.. 91

Figure 67: Most participants preferred EMS to no-EMS interface condition. 91

Figure 68: Participant balancing the marble (with consent of the participant).............. 92

Figure 69: EMS application areas follow research areas in mechanical actuators........ 94

Figure 70: Using the sense of proprioception alone to rewind a video, while maintaining eye contact with his audience. 97

Figure 71: Symmetric proprioceptive interaction revolves around the pose of one of the user's limbs, here the wrist. 98

Figure 72: In an asymmetric proprioceptive interaction, input and output occur through different limbs.. 98

Figure 73: The main components of pose-IO.. 99

Figure 74: Using pose-IO to scrub a video, i.e., one's wrist as the I/O device. 99

Figure 75: Red-hands for one: under control of pose-IO, the user's right hand tries to slap the user's left hand. The goal is to evade getting slapped by his own hand.......... 101

Figure 76: Imaginary pong for one: (b) The computer opponent plays the imaginary ball left by flexing its hand (c) The user returns the shots by moving his hand to where he feels the ball went. 102

Figure 77: The 3D printed pose-IO bracelet.. 103

Figure 78: The pose-IO bracelet unrolled and with the top enclosure removed.......... 103

Figure 79: EMS electrode placement for wrist flexion and extension. 104

Figure 80: Behavior of pose-IO's PID control. 105

Figure 81: The pose-IO version based on EMG uses hand-overextension as invocation gesture.. 105

Figure 82: EMG electrode placed on wrist extensor and flexor................................. 106

Figure 83: Apparatus: a barrier prevented participants from seeing their wrist.......... 107

Figure 84: Differences in angle between target pose and recreated pose.................... 108

Figure 85: (a) Wrist actuated by pose-IO. (b) Pose recreated by participant. 108

Figure 86: All participants rated EMS red hands as fun (1: uninteresting, 5: fun)....... 110

Figure 87: Participants playing EMS red hands (image with participants' consent).... 111

Figure 88: (a) This spray can needs to be shaken before use. (b) Affordance++ allows the spray can to make the user shake it before use. It implements this by electrically stimulating the user's muscles. (c) The can is "willing" to be used. 114

Figure 89: Here, affordance++ helps the user interact with an unfamiliar lamp. 115

Figure 90: Example of a tool that requires a correct order of execution. 116

Figure 91: Cutting the avocado in half with this tool. 116

Figure 92: Extracting the pit using the tool. 117

Figure 93: This tool allows for slicing the avocado without peeling the skin. 117

Figure 94: Changing behavior according to the object's state: (a) this cold cup affords grasping, (b) when hot it repels the user's hand, but (c) its handle continues to afford grasping. 118

Figure 95: A door that repels using affordance++. 118

Figure 96: While approaching a door with the intention to knock, affordance++ allows the gesture amplitude to be communicated as a function of distance from user to object. 119

Figure 97: Overwriting the user's intention by providing the right turning direction.. 119

Figure 98: This affordance++ prototype uses an *Optitrack* setup for tracking. 120

Figure 99: Electrode placement for each gesture used in affordance++. 121

Figure 100: (a) Here the user approaches the designer lamp. (b) In Unity3D, we detect the collision with the bounding volumes and trigger the pulling down gesture. 122

Figure 101: Wearable prototype based on RFID. 122

Figure 102: Four participants exploring (a,b) *repel* and (c,d) *raise*. 123

Figure 103: Correctly guessed behaviors, average for all users. 124

Figure 104: Confidence per behavior (Median and IQR, and total range of variation). 125

Figure 105: Four participants exploring (a,b) *shake* and (c,d) *rotate*. (b) This *Rotate* failed because the wrist flexed more than it turned. 125

Figure 106: Four participants exploring (a, b) *drop* and (c,d) *squeeze*. (b) Here the muscle actuation was not sufficient to cause the participant to *drop* the object........... 126

Figure 107: The objects used in Task 2. 127

Figure 108: Participants' ratings for the six test objects. 128

Figure 109: Example of overriding the conventional behavior. 129

Figure 110: An interactive wind tunnel simulation with pen input and output, which is based on EMS. 132

Figure 111: Selecting which sketches to simulate in the wind tunnel. 133

Figure 112: (a) The user defines a cross section of the streamlines. (b) Then, draws two axes of an X-Y plot, and (c) drags along the X-axis while muscle-plotter actuates the wrist to plot the Y values. 134

Figure 113: The user inspects a region of the wind speed charts by zooming. 134

Figure 114: To obtain the drag coefficient, the user (a) writes the command and sketches a vertical line, and (b) selects the desired car with a pigtail. Then, (c) our system outputs the value as a tick-mark to the right as the user traces the line. 135

Figure 115: (a) Plotting a fifth degree polynomial and (b) finding out via a query where the integral of this polynomial equals 2 units. 135

Figure 116: Iterating through the design of a RC filter by scribbling the old value a user can input a new capacitor value and then re-plot the filter response. 136

Figure 117: Simple form widgets for quick queries in muscle-plotter: (a) a multiple checkbox and (b) a radio button. 136

Figure 118: Fitting a linear regression in a bar chart. 137

Figure 119: Using muscle-plotter to explore how rays of light refract through convex and concave lenses. 137

Figure 120: Electrode placement to perform (a) pen right, pen up, (b) pen left, and pen down (i.e., “brake”) movements. 140

Figure 121: Average pulse width per slant, for both extensor and flexor muscles, for all participants of our experiment. 141

Figure 122: Key principles in muscle-plotter’s control loop. 142

Figure 123: One trial: one plot (image from the study). 145

Figure 124: Setup for our experiment (image taken from the study). 146

Figure 125: Raw data from 8 participants and 128 trials. The white trace represents the desired trajectory and colored traces depict user drawn plots (scale in mm). Average per function represents error (distance to target). 146

Figure 126: Fourier transforms: original signal (red) and users output (green). 147

Figure 127: Average error (in mm) for each trial in relation to its duration (in seconds). Each color is a different participant. 148

Figure 128: Extending proprioceptive interaction to other parts of the body: (a) the feet and (b) the upper arm. 150

Figure 129: (left) While devices based on mechanical actuation *add* mechanical components to the user’s body, (right) systems based on electrical muscle stimulation instead *borrow* the user’s skeleton and muscles (Exoskeleton image source: Université Libre de Bruxelles; used with permission.) 152

Figure 130: A possible evolution of interactive devices. (Images in CC0 Creative Commons by *JESHOOTScom*, *Free-Photos* and *geralt_*). 153

Figure 131: Ad Infinitum, a parasitical machine that lives off human energy. 156

1 INTRODUCTION

A key question that has driven interactive computing for decades is: How can interactive devices connect with users in the most immediate and intimate way?

When we think back to the early days of computing, user and device were quite distant, often located in separate rooms. Then, in the late 1970s, personal computers "moved in" with users. In the 1990s, mobile devices put computing into users' pockets. More recently, wearables brought computing into constant physical contact with the user's skin. These transitions proved to be useful: moving closer to users and spending more time with them allowed devices to sense more of the user. This knowledge allowed interactive devices to act more personal.

The main question that drives our research is: what is the next logical step? How can computing devices become even more personal?

One way of investigating this is to ask where future interactive devices will be located with respect to the user's body. Some researchers argue that the next generation of interactive devices will move past the user's skin, and be directly implanted inside the user's body. This has already happened in that we have pacemakers, insulin pumps, and cochlear implants, all of which are located inside the user's body. However, we argue that what we see is not devices moving towards the inside of the user's body but rather towards the biological "interface" of the user's body they need to address, in order to perform their function, i.e., the heart, the ear, and so forth; Figure 1 illustrates this point.

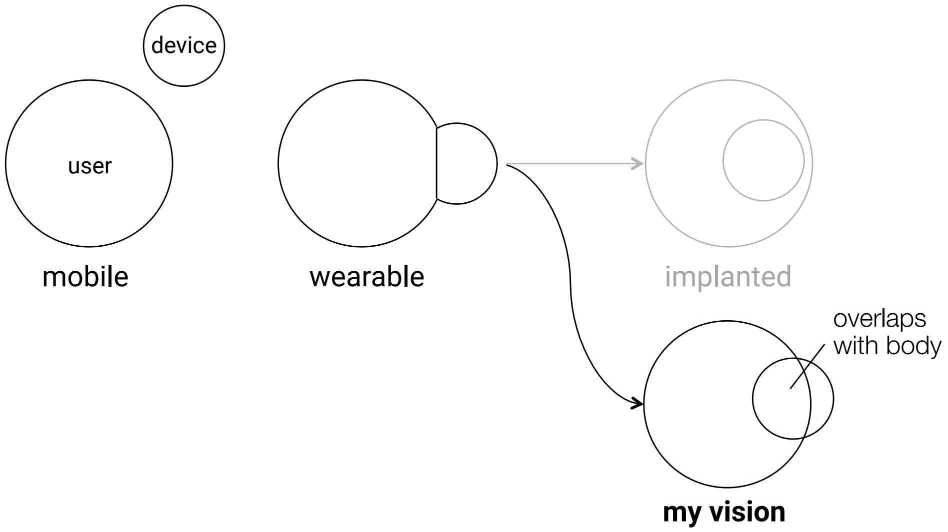

Figure 1: Our proposed interactive systems borrow parts of the user's body, making the resulting devices effectively overlap with the user's body.

This idea that devices are moving towards the biological interface holds the key to a more immediate and personal communication between device and user. To further advance this path, we created a set of devices that intentionally *borrow* parts of the user's body for input and output, rather than adding more technology to the body.

In this dissertation we present one specific flavor of such devices, i.e., devices that *borrow the user's muscles*. We engineered input/output devices that interact with the user by reading and controlling muscle activity. To achieve the latter, we built our devices based on medical-grade muscle stimulators: these send electrical impulses to the user's muscles via electrodes attached to the user's skin. These pain-free and medically safe impulses cause the user's muscles to contract involuntarily.

While electrical muscle stimulation (EMS) devices have been used to regenerate lost motor functions in rehabilitation medicine since the 1960s, in this dissertation, we propose a new perspective: EMS as a means for creating interactive systems.

1.1 An example of an interactive system based on EMS

Before we dive any deeper, let us start with a simple example of one of our EMS-based interactive systems; a device that we called *pose-IO* and is depicted in Figure 2.

The Pose-IO device consists of a self-contained bracelet that users wear under their clothing. It drives two pairs of electrodes, which are attached to the user's forearm muscles: one pair of electrodes attaches to the muscle that flexes the wrist and the other pair to the muscle that extends the wrist.

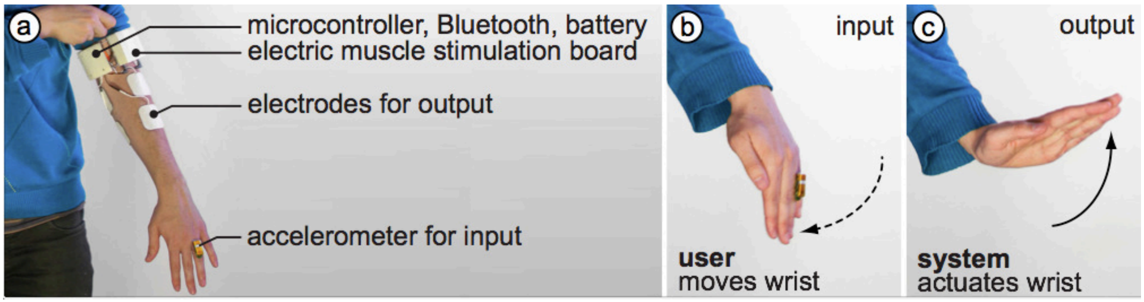

Figure 2: An interactive system based on electrical muscle stimulation: pose-IO.

Pose-IO allows users to control a single variable, such as the position of a video playhead. As shown in Figure 2b users can *set* the position of the playhead by posing their wrist, which the device senses using a 3-axis accelerometer. At the same time, as shown in Figure 2c, the system continuously *writes* its output into the user's wrist posture, i.e., as the video continues to play, users find their wrist bending further and further upwards; pose-IO achieves this by actuating the user's wrist using EMS.

1.2 Our contributions

Our contribution is the creation of interactive systems based on reading and writing to the user’s muscles, allowing for both input and output, respectively. While wearables that leverage the user’s muscles for input are well understood (e.g., motion sensors, EMG, etc.), the converse—wearables that leverage the user’s muscles for output—is, unfortunately, still an open challenge. We tackle this challenge by proposing electrical muscle stimulation as a way to write to the user’s muscles.

Our proposal of using EMS to create interactive systems results in two benefits for the field of human computer interaction: (1) a new type of input/output device that can be operated eyes-free, since it is based on reading and writing to the user’s muscles. (2) A technical approach to miniaturizing force feedback devices. The latter sheds new light in the elusive challenge of creating strong but small haptic machinery capable of actuating the human body. We show that our resulting EMS-based force feedback devices are capable of producing realistic haptic feedback while being sufficiently small to be worn under the user’s clothing.

1.3 Two lines of exploration

Our interactive systems based on EMS form two main categories, which are depicted in Figure 3: (1) devices that increase the user’s sense of immersion in virtual reality by simulating large forces, such as wind, physical impact, or walls and heavy objects; and, (2) devices that allow users eyes-free access to information by means of their proprioceptive sense, such as reading/writing the value of a variable in a computer system, accessing information about a tool, or interacting with data plots. Additionally, we will also describe an artwork in chapter 4 that helped us in exposing and debating the societal implications of these technologies.

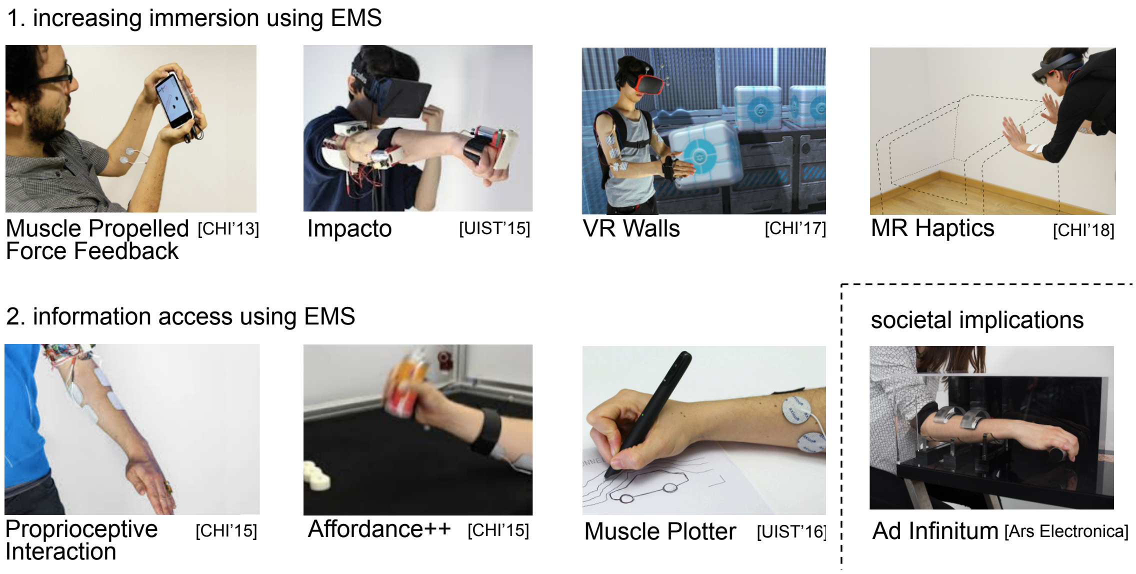

Figure 3: Our two lines of work in interactive systems based on EMS.

1.4 Structure of this dissertation

This dissertation is organized in five chapters. After this introduction (chapter 1), we take the reader into a review of the prior work that we will build on, with a particular emphasis on the field of haptics (chapter 2). The two subsequent chapters are dedicated to our two lines of work: increasing immersion using EMS (chapter 3), and information access using EMS (chapter 4). We conclude this dissertation by summarizing, discussing and comparing our approach with state-of-the-art wearable haptic devices, followed by outlining the open technical, ethical and philosophical challenges for this emergent field (chapter 5).

2 RELATED WORK

The work presented in this dissertation builds primarily on the fields of haptics, wearable computing and electrical muscle stimulation. In this chapter, we overview prior work that addresses both the tactile and proprioceptive sense, eyes-free interaction, force feedback, muscle sensing, and present the working principles behind electrical muscle stimulation.

2.1 Haptics

The field of haptics is concerned with creating devices that simulate the properties of objects that “we can feel”, i.e., haptic properties. These are the properties of objects that are not perceived with either the auditory or visual apparatuses, but instead, are sensed via receptors in our skin, tendons and muscles [53]. An example of a mundane, yet crucial, haptic property that we feel is the *texture* of a surface (e.g., is it smooth, rough, or does it encode a braille message?).

The field of haptics is generally subdivided into two sub-areas: *tactile feedback* (i.e., feeling a surface, feeling its texture, etc.) and *proprioceptive feedback* (i.e., feeling forces associated with an object, such as its weight) [53]. The work in this dissertation is concerned with creating an alternative technological apparatus for proprioceptive haptics (forces). Still, for the sake of completeness and theoretical framing, we start with a brief overview of the prior work on tactile haptics.

2.1.1 Tactile feedback

Most haptic devices that stimulate our sense of touch aim at creating the illusion that we made contact with an object. To exemplify, imagine stretching your arm in virtual reality and tapping on a virtual wall; then, by moving your finger across the wall you would also expect to feel its texture (and its temperature but that is outside of our scope).

Perceiving touch. Before we survey the techniques used to create the illusion of contact with a virtual object, let us briefly survey the neural mechanisms of our sense of touch: we feel that we have touched something, or that something has touched us, by means of specialized nerve cells, called mechanoreceptors, which are lodged in our skin. These cells are able to transform a mechanical input, such as the pressure applied to our skin, into an electrical signal that is sent to the brain, where it is processed into what we feel as “a touch”.

Mechanoreceptors that deal with sensing the tactile qualities of our environment exist in a wide variety (heavily simplified from [41]): the Meissner corpuscles, which detect light touch and adapt to changes in texture (around 50 Hz); the Ruffini corpuscles, which detect tension deep in the skin and fascia; the Merkel nerve endings, which detect sustained pressure; and, the Pacinian corpuscles that detect rapid vibrations (between 200 –300 Hz). Therefore, to create the illusion of touch in virtual environments, such as when we reach out and touch a virtual wall, researchers rely on stimulating these receptors using haptic devices.

Vibration. In order to achieve the illusion of contact the majority of low-cost commercial haptic devices rely on small vibration motors, such as those found in contemporary handheld VR controllers. The relatively low costs associated to these vibration motors allowed researchers to build devices with many vibrotactile units. For instance, Lindeman et al. simulate impacts in virtual reality using a suit that contains 16 vibrotactile actuators [99]. In their virtual reality "shooting" application, the suit communicates the spatial location of the shots fired at the user by activating the respective vibrotactile cell.

Vibration quickly became the most common haptic feedback modality in VR because it allows for wearable form factors, such as gloves (e.g., the *CyberTouch* by Virtual Technologies [20]) and vests [99]. Unfortunately, as evidence suggests, vibration feedback becomes harder to perceive when users are in motion [78]. Also, when crosschecking with the list of mechanoreceptors found in our skin, we find that a vibration-based haptic device will only stimulate two out of available four receptors, i.e., it stimulates only the Meissner and Pacinian nerve cells [89].

Pressure. In order to create more realistic sensations of touch, researchers utilize devices that press against the user's skin. These devices will also activate also the Ruffini and Merkel cells, which are sensitive to pressure [89]. Pressing against the user's skin, especially at the finger pads, increases realism when manipulating/grasping virtual objects.

Researchers have explored many technical avenues to create pressure at the user's fingertips, such as pneumatics or jamming. Pneumatic Gloves include air pockets that inflate when the user's fingertips touch a virtual object [12]. The *Teleact* glove [137], for example, featured 30 air pockets around the user's fingers and palm. This technology is also used in surgical manipulators to emulate the experience of touching soft tissue [90]. Another example is *Wearable Jamming Mitten* [138], which locks the user's hand in a clenched position when the user grasps a virtual object. It achieves this by jamming, i.e., by removing all air causing the substance inside the glove to interlock.

Combining vibration, pressure and skin stretch. Real world interactions with objects simultaneously elicit a wide range of sensations, including pressure, vibration but also skin-stretching. In order to achieve more compelling simulations that allow users to interact with objects, researchers have combined all of the previous approaches in consolidated haptic devices. For example, the *HapThimble* [80] is a finger-mounted haptic device that emulates contact with virtual objects by pushing a vibrotactile actuator against the user's fingertip—effectively combining vibration and pressure information. Other devices incorporate also skin stretch actuators that are used for simulating directional forces when contacting objects [26, 16]. These haptic devices laterally stretch the user's finger pad skin using actuators, such as miniaturized Stewart platforms or by pulling cables [15].

Non-wearable methods for tactile feedback. It is also worth noting that tactile stimuli can be transmitted through the air around the user using ultrasonic beam forming [68] or air vortexes [141]. Instead of attaching hardware to the user's body, these approaches require stationary emitters positioned in front of the user. Therefore, such devices are prone to occlusion and restrict the interactive space to small volumes (e.g., one meter in most cases).

2.1.2 Proprioceptive feedback

We ask you, again, to imagine the hypothetical scene from before: reaching your arm to touch a virtual wall. In real life, there is one additional haptic cue that you feel when you touch a wall: its force. In fact, a wall perfectly illustrates your ability to feel forces since the wall pushes back against you as much as you push onto it (Newton's 3rd law, i.e., every force has an equal counter-force). Devices that are concerned with the proprioceptive, rather than tactile, side of haptics are designed to render the sensation of forces.

Perceiving forces. We perceive forces by means of our proprioceptive sense—proprioception allows sensing the position, orientation, and movement of our limbs, joints, and muscles [29]. Proprioception works by combining the information of mainly three mechanoreceptors: the muscle spindles (receptors lodged in our muscle fibers), which detect changes in the length of the muscle; the Golgi tendon organs (receptors at the interface between muscles and tendons), which sense changes in muscle tension; and, lastly, sensors inside the joint capsules, which sense the joint's rotation [40, 128]. Research suggests that our proprioception can also operate independently from the visual and auditory channels [29, 40]. Coming back to our "touching a wall" example, this means you can feel the force that the wall exerts on you even with your eyes closed.

Note that some authors prefer the term *kinesthetic* feedback (which evokes the idea of motion) rather than proprioceptive feedback. For the sake of the argument of this thesis, which deals directly with the proprioceptive sense, we opt for proprioceptive feedback instead.

Besides the research in the medical domain (psychophysics and neuroscience), proprioception has been approached from the perspective of human computer interaction—when a user's body is actuated by haptic machinery, the user feels the resulting forces via their proprioception. We now review the different technical apparatuses that researchers created for actuating the user's limbs with force feedback.

2.1.3 Motor-based force feedback

Force feedback devices mechanically administer forces to the user's body. Typically, these devices generate and apply forces using robotic arms [103], pulley systems [107] and exoskeletons [152].

Robotic arms. One way to actuate the user's limbs is to use strong robotic arms for force feedback, such as in the system by Yokokohji et al. [158]. Similarly, Gruenbaum et al. leverage an industrial robotic manipulator as a stand-in for a control panel of a virtual automobile [46].

Pulley systems. A canonical example of a pulley-based haptic device is the *SPIDAR* [107], which displaces the user's fingertips by pulling tethers that are attached to rings worn on the user's fingers. These tethers are pulled against a stationary frame by motors. Variations of this popular SPIDAR design have been used in CAVE-like simulators with force feedback, such as a boxing simulator [57] and a catch-ball simulator [75]. Lastly, *SPIDAR-W* [108] is based on the same SPIDAR principle but, instead of being stationary, is mounted into a 1.5 x 1.5 x 1.5 m cage that the user carries around.

Exoskeletons. Much like our last example (SPIDAR-W), exoskeleton-based devices require that the complete hardware (tethers, frames and motors) be mounted on the user's body. Exoskeletons are typically constructed from several rigid linkages connected to each other by moving joints, which are then actuated using strong motors. In an exoskeleton, the linkages are attached to the human body, typically using straps. *SAM* [93] is an example of a portable exoskeleton that is able to actuate the user's arm in 7 DOF (i.e., 7 degrees of freedom, meaning this device has seven moving joints). An early example of using exoskeletons to increase immersion is the *Dextrous Hand Master* [64], which is an exoskeleton glove that provides force feedback to the user's fingers in VR. Other iconic exoskeletons used in VR are, for instance, *Dexmo* [47], which is a more recent rendition of a glove exoskeleton that uses lighter motors that can only brake; *EXO-UL7* [131], a stationary 7 DOF exoskeleton for both arms; or *FlexTensor* [152], a minimal exoskeleton that only actuates the biceps.

Combining with vibration. Researchers have also combined force-feedback devices with tactile stimulation: Kron et al. combined a force feedback hand exoskeleton with an eccentric motor on each finger [87]; similarly, *the Data Glove* by VPL [37] and *Cyber Glove Force* by Kramer et al. [86] do provide hybrid force & tactile sensations.

Weight-force tradeoff. The tradeoff between a motor's output force and its weight is the crucial limiting factor of force feedback devices such as exoskeletons. To give the reader a visual analogy, we compare two previously mentioned exoskeleton devices in Figure 4: (a) The *EXO-UL7* [131] outputs forces closely matching a human's strength because its motors provide 2.2 Nm/kg of torque. Unfortunately, a device with this output power/weight is no longer portable and has to be affixed to a wall. (b) The *SAM* [93] exoskeleton was optimized to achieve maximum portability and maximize each joint's torque; the resulting device weights 6 kg without batteries (or 12 kg for both arms) but its output power is only 1/20th of a human maximum's torque per joint.

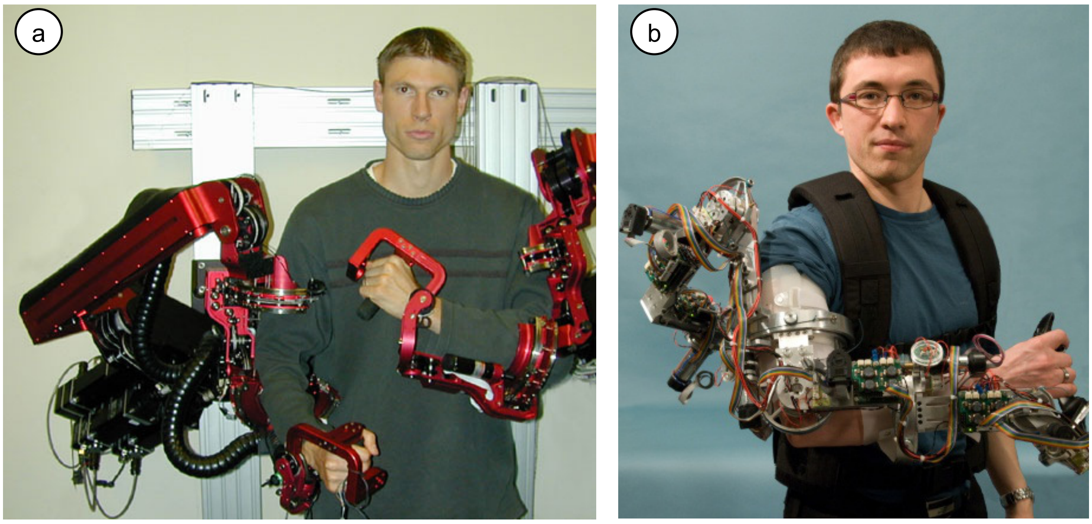

Figure 4: Illustrating the weight-force tradeoff in exoskeletons by comparing (a) the stationary EXO-UL7 and (b) the portable SAM exoskeleton (images from UCLA Bionics Lab press kit and with permission of Université Libre de Bruxelles).

Through this analogy we see that, even when optimized for weight, exoskeleton devices do not play along with today's requirements for VR devices: mobile and wearable. As we will address later in Chapter 3, we took this limitation of traditional haptic devices as a starting point to our work on increasing realism using electrical muscle stimulation.

2.2 Wearables based on Proprioceptive Input

The second area that we cover in this related work chapter encompasses the field of wearable computing. In particular, we are interested in wearable devices capable of tracking the user's movements and muscle contractions.

2.2.1 Wearables with motion tracking (IMUs and computer vision)

Today, many commodity devices are able to track the user's motion using internal measurement units (IMUs), which are circuits that measure motion using a combination of accelerometers, gyroscopes and sometimes also magnetometers. Popular examples are all around us: our smartphones use their IMU to determine the screen's tilt, while our fitness bracelets use accelerometers to count steps.

Besides IMUs, researchers also rely on computer vision on wearables that track the user's lims. For instance, *Cyclops* [24] uses a fisheye camera attached to the user's chest, which sees the user's whole body, allowing it to track the user's limbs. Similarly, *OmniTouch* [55] uses a Microsoft Kinect depth camera to track when the user's fingers touch the user's forearm. Likewise, *Digits* [81] is a wrist-worn camera combined with an infrared line projector that track the user's finger positions. More recently, small high frequency radars, such as *Soli* [155] started to be used for tracking the user's fingers.

2.2.2 Wearables that read muscle activity (EMGs and MMGs)

While the previous approach is particularly good at tracking the motion of a limb that the sensor is attached to (e.g., your *fitbit* bracelet tracks your wrist motion), these devices cannot determine the strength of a muscular contraction, i.e., neither a camera nor an IMU can determine how much force you are exerting when you are moving.

To measure the force of a muscle contraction researchers rely on measuring the muscle's activity. This can be achieved in a myriad of non-invasive methods; however, we focus on two methods that can afford a wearable form factor: electromyography (EMG) and mechanomyography (MMG).

MMG is a technique that relies on listening to the vibrations produced by the onset of a muscular contraction [144]— the sensor employed in MMGs is a microphone. The idea of using wearable microphones to capture the sounds arriving at the user's skin has been used for inferring the user's muscular contractions [22] but also to infer what/where the user is touching (*sound of one hand* [2] and also *Skinput* [54]).

EMG is a technique that directly amplifies the electrical signals, inherent from the nervous system, which regulate our muscular contractions [148]. In its non-invasive form, called surface electromyography (sEMG), pre-gelled electrodes are attached to the user's skin. These capture the residual electrical potential from neuronal firings that can still be measured through the skin. In the field of human computer interaction (HCI), these EMG devices became a popular way to enable what Felzer et al. described as "hands-free control applications" [36]. For instance, Saponas et al. used EMG to create wearable input devices, such as a music player that users can control eyes-free by means of muscle contractions, even while jogging [134].

The conceptual outcome of EMG and MMG-based devices is that they turn the user's muscles into actual input devices. Saponas et al. highlight that the biggest benefit in using the muscles as input devices is that the body is an "always available" resource [134].

While EMG has mostly been used for medical or research purposes, it was not until recently that a commercial product came along. The *Myo* [6] is a bracelet that combines EMG (dry electrodes rather than sticky ones) and IMUs, allowing users to contract their muscles as input for games and to control their media player applications.

Also, it is worth noticing that the signals captured in high-quality MMG recordings have been shown to reliably depict the intensity of a muscle contraction, even when compared to EMG recordings of the same subjects [148].

2.2.3 Eyes-free interaction

The sensing technologies we just introduced (e.g., EMG, cameras, IMUs) have been instrumental in assisting researchers in creating ways for users to input information even when they are not looking; this is the so-called *eyes-free interaction*.

Eyes-free input. There are several examples of devices that allow eyes-free input, such as the EMG wearable device by Saponas et al. [134] that we covered before. Another device, the *Pinstripe* [77], adds sensing to the user's clothing by embedding conductive threads in it; this enables the garment to sense 1D gestures designed to be easily performed without relying on the user's visual apparatus. *Imaginary Interfaces* [48] are also non-visual input devices that enable also an additional 2D spatial component: users can operate them eyes-free by touching the palm of their non-dominant hand because the touch elicits tactile and proprioceptive senses [49]. *Imaginary Phone* goes one step further and instantiates an input-device using the human hand, which results in a small acquisition time because there is no need to reach for the mobile device in the pocket [50].

Eyes-free output. For an interactive dialog to take place in an eyes-free manner, users need not only eyes-free input, but also output. The predominant output modality for eyes-free output is auditory input, which is popular especially for interactive appliances such as digital home assistants (e.g., Google Home and Amazon Alexa). The auditory modality is however limited to quiet environments and to situations in which the user is not engaged into conversations. To circumvent the limitations of auditory output, researchers instead build eyes-free output devices using tactile feedback. For instance, Tan et al. utilized a 3x3 vibrotactile actuator matrix as a wearable eyes-free output device designed to communicate messages to the user's tactile sense [146]. Unfortunately, vibration is also subject to a range of limitations. Firstly, as we mentioned before, the recognition of vibrotactile patterns degrades in mobile situations, such as when walking [78]. Furthermore, vibrotactile output was found to offer limited bandwidth [109] and is hard to learn because it lacks mnemonic properties [94].

One possible solution would be to create wearables that provide proprioceptive cues (i.e., that actuate the user's muscles) as an eyes-free output modality. However, as we have discussed, the constraints of exoskeleton-based devices sever this possibility. As we will cover in Chapter 4, we took the lack of more expressive modalities for eyes-free output as a starting point for our work on information access using electrical muscle stimulation.

Closing the loop between eyes-free input & output. A device that enables a full eyes-free interaction (i.e., input and output) is *Gesture Output* [132]. This device, which we co-authored during this dissertation, is a modified smartphone with an actuated transparent overlay onto the screen. If the user's finger rests on the screen, the device can actuate the finger by means of pulling the overlay with motors; this results in the user's finger moving, involuntarily, in 2D shapes. This output modality allows *Gesture Output* to communicate spatial gestures, which the user can sense eyes-free by means of their proprioceptive sense. Simultaneously, using the touchscreen that is native to today's smartphones, the device also accepts gestures as input. This allows the input and output languages to be symmetric [132], i.e., they share the exact same alphabet of gestures. While *Gesture Output* is not a wearable device, since it is deployed in a smartphone, yet it is precisely this idea of a unified I/O space that is entirely perceived by the user's proprioception that we will echo in our *proprioceptive interaction* paradigm (Chapter 4).

2.3 Electrical muscle stimulation

To conclude, we introduce electrical muscle stimulation (EMS), which is a technique that applies electrical impulses to the user's muscles in order to involuntarily contract them [100]. The basic concept of stimulating muscles by means of electricity goes back as far as 1791, when Galvani discovered that electric current caused contractions on the leg muscles of an inanimate frog [157]. The development of the technology that is found in today's EMS devices spawns from advancements in rehabilitation medicine that began in the 1960s [27, 145]. Since then EMS has become an effective technique in clinical rehabilitation to regenerate lost motor functions, such as in patients with limited mobility due to spinal cord injury.

2.3.1 Working principle

EMS devices deliver impulses by means of electrode pairs formed by an anode and a cathode [1]. When the electric current that travels from cathode to anode it crosses nerves and motor neurons and adds to the neurons' internal voltage [100]. When the resulting voltage surpasses a certain threshold, it causes the motor neurons to "fire" [41]. This signal, called the *action potential*, travels down the nerve until it reaches the muscle fibers and causes them to contract.

EMS stimulators can achieve varying levels of muscle contraction by modulating the amplitude of the current. Higher currents generally penetrate deeper into the user's skin and tissue, where they reach additional neurons and can thus innervate more muscle fibers [100].

The majority of EMS-based systems use pre-gelled electrodes that attach to the user’s skin; this gel acts as both glue and as an impedance matcher [1]. These electrodes are suitable for time periods of up to a few hours per session at which point the gel layer needs to be replenished. Implanted microelectrode arrays [18] allow for longer use at the expense of requiring surgical insertion. These are only used mostly in medical research and in rare rehabilitation cases where the user’s motor functions are permanently damaged.

This simple working principle generated several flavors of the technique and each has its own acronym, such as FES, TENS, TES, or NMES [100]. Functional Electrical Stimulation (FES), for example, denotes the usage of electrical stimulation primarily to restore or improve the function of paralyzed or affected muscles [1]. In clinical trials, FES has been shown to improve muscle performance for patients with decreased motor function without long-term side effects [60], the downside being the temporary discomfort caused by the “tingling sensation” while using the device [23]. These acronyms mostly denote different stimulation parameters, stimulation protocols, or application areas [100]. In this dissertation we adopted the term EMS, as it is a more encompassing term.

2.3.2 EMS signal generator

The typical way of driving an EMS device is to use one circuit for the logic (which can be as simple as an Arduino microcontroller) that drives a separate medical-grade EMS signal generator. This signal generator outputs alternating pulses (a positive pulse followed by a negative one) that are either sinusoidal or square-shaped [76] usually at a frequency between 15 Hz – 120 Hz with a pulse duration of 50 – 400 µs. EMS signal generators are current-limited, thus preventing the device from ever generating currents either above the safety level (100 mA), or above the calibrated intensity. It is worth noting that there is a non-linear relationship between the intensity and the duration of a stimulation pulse, which is known as *strength-duration* curve [82]. This curve describes how these two parameters are interchangeable; in short, the longer the pulse duration the less current is needed to trigger the contraction. Further safety features of muscle stimulators include detecting whether the electrodes are attached to the user and stop the system otherwise.

2.3.3 Deploying an EMS system

In EMS, two electrodes are attached to the user’s skin: one electrode is attached near the base of the muscle, closer to its tendons, and a second electrode is attached to the muscle belly, typically in the middle of the muscle, where most of the fibers are.

Figure 5 shows the placement of two electrodes to stimulate the user's biceps muscle, and the effect of a stimulation with a current of 17 mA and a pulse width of 200 µs for a period of 300 ms. Typical muscle contractions start around 10–50 ms from the stimulation pulse onset [23]. Finding these anatomically correct locations requires a certain amount of expertise. While locations along the arm, shoulder, and legs are relatively straightforward, some set-ups are inherently taboo such as electrodes across

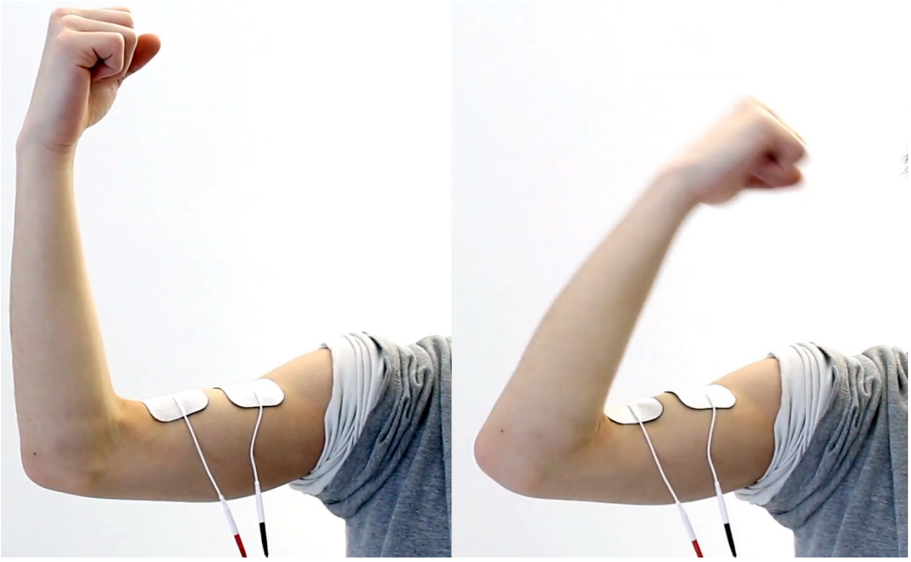

Figure 5: Electrically actuating the biceps muscle using EMS.

The current levels required to actuate a limb vary across users and muscles. To assure pain-free operation, one typically calibrates the device by starting at 0 mA and increases the current from there. The experimenter then registers the lowest current that causes some movement of the muscle, and then users set the highest current that they still feel comfortable to them; the range in between is what the device can use to generate different levels of contractions.

2.3.4 Application areas of electrical muscle stimulation

EMS in medicine. As mentioned earlier, the original application area of EMS was and continues to be replacing/supporting human motor functions in the field of rehabilitation medicine. Here, EMS promises to help patients who lost motor functions, e.g., as a result of spinal cord injury, to regain control over the otherwise inaccessible limbs. Most of the applications artificially induce body movements, such as grasping, walking, swallowing, mitigating the drop-foot syndrome, and assisting in standing upright [121]. This application space makes excellent use of the qualities of EMS and is probably the closest a patient can get to restore the body as it was prior to the injury.

EMS in Art. The first non-medical applications of EMS came from the artistic domain, as artists in the 1980s started to explore EMS as part of interactive art installations. The early works of Stelarc depicted examples of an initial form of teleoperation between a human arm (moved by EMS) and a robotic arm [120]. Also pioneer was the work of Elsenaar, in his lecture-performances a computer program narrated to the audience while a performer's facial muscles were stimulated [32]—the performer's face, which was actuated by means of EMS, provided a surrogate face to the computer. Similarly, the performances of Daito Manabe demonstrated the transfer of facial expressions between two users; one user was instrumented using sensing electrodes (EMG), while the other was provided with electrodes for stimulation [101].

EMS in HCI. Only more recently, circa 2003, was EMS used for interactive purposes. Figure 6 depicts a chronologically arranged timeline of EMS research in the field of Human Computer Interaction.

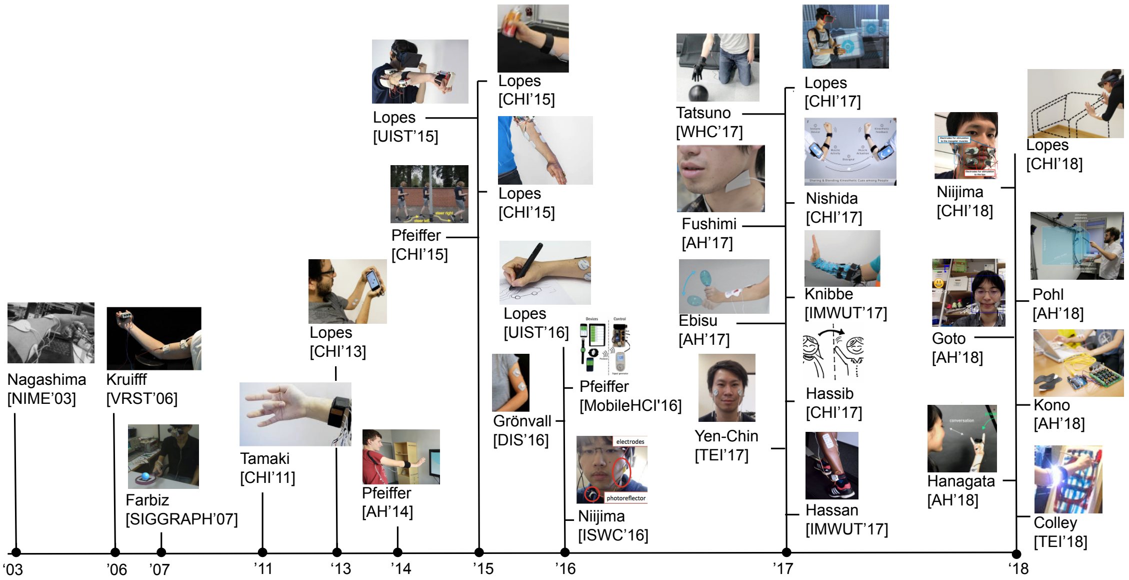

Figure 6: Timeline of EMS research in the field of Human Computer Interaction. (To preserve visual clarity we do not display publications from adjunct proceedings or other non-archival formats, except the early work by Farbiz et al.).

These interactive devices based on EMS, engineered by researchers in HCI, appear clustered around four main themes, which we will describe next: (1) **training**, (2) **immersion**, (3) a rudimentary form of **teleoperation**, and (4) **information access**.

1. Tutorial/Training. These projects employ EMS as an actuation system that guides users in performing physical tasks. For instance, *PossessedHand* [147] applies EMS on the user's forearm muscles to guide the user into the correct finger poses for learning how to play a new string instrument. The wearable form factor of EMS systems also makes them attractive for mobile applications. For instance, *Pedestrian Cruise Control* [123] provides walking directions (left vs. right) to the wearer by using EMS to turn their legs into the desired direction. Similarly, *FootStriker* [58] uses EMS to correct a runner's heel placement while their foot is in midair, prior to taking the next step. Tatsuno et al's system [149] utilizes EMS to correct the arm rotation during a bowling swing. *Vibr-o-matic* [38] stimulates muscles of the abdomen and larynx to aid novice singers to perform vibrato (amplitude modulation). Also, in the domain of music tutors, Ebisu et al. utilized EMS to assist novice musicians in keeping the rhythm—their system stimulates the user's wrist muscles as they hold maracas [31]. Our *affordance++* system, described in chapter 4, allows users to learn how to operate objects they might not be familiar with— it allows objects to communicate their affordance by means of actuating the user's muscles to perform the correct manipulations. A spray can enabled by affordance++, for example, may compel the user's hand to shake the can prior to its use. Finally, Niijima et al. utilize EMS on the user's facial muscles to create involuntary teeth clenching that promotes a stronger voluntary contraction of the upper arms (i.e., this takes advantage of the fact that we naturally clench our teeth when weight lifting) [110].

2. Immersion. These systems use EMS to provide users with force feedback. This line of research started with Kruijff et al. providing EMS induced force feedback for users playing a video game on a desktop computer [88]. While traditional vibrotactile feedback can highlight in-game events with vibration, EMS is able to produce directional forces. The fact that EMS miniaturizes well, allowed us to apply such feedback to our work with mobile devices, as in *muscle-propelled force feedback*, which we describe in the next chapter. These, EMS-driven physical sensations increase the realism of interactive virtual environments. For instance, Farbiz et al. used EMS on the wrist muscles to render the sensation of a ball hitting a racket in an augmented reality tennis game [34]. Pfeiffer et al. applied EMS to the user's wrist to increase the realism of the sensation of grabbing a soft or hard object, while interacting in midair with a large-scale display [122]. Niijima et al. stimulate the user's masseter muscles (jaw) to re-create the resistance of chewing virtual food [111]. In chapter 3, we demonstrated how to use EMS force feedback to simulate punches of a virtual boxer (in *impacto*), as well as to simulate the forces of walls and heavy objects in both virtual reality (in *VR Walls*), and mixed reality (in *EMS in mixed reality*).

3. Teleoperation. Recently, researchers started using EMS to render the movements of a remote user; this can be thought as a rudimentary form of teleoperation since, currently, the precision of EMS actuators is still limited (more on this in chapter 3). One example of this is *bioSync* [112], a wearable EMS-based system that allows one user to transfer their muscle contractions, sensed via EMG, to a remote user. Similarly, *Paralogue* [51] renders the head direction of a remote user by using EMS on the user's wrist muscles, i.e., the local user's hand acts as a proxy of the remote user's head and, when the remote user's head tilts or turns, the local user's hand tilts and turns in response.

4. Information Access. Our *pose-IO*, which we describe in chapter 4, is wearable device that allows users to set and retrieve the value of a variable in a computer system; users retrieve the value by finding their wrists posed by means of EMS. Similarly, other systems provide users a way to receive information by feeling their muscles moving. For instance, *Feel Radio* [45] allows users to feel wireless traffic such as Wi-Fi or Bluetooth by means of muscle contractions. Also, *Emotion Actuator* [59] allows a user to feel an EMS-induced gesture that maps to an emotional state of another person. The emotional state of the remote person is sensed by means of electroencephalography (EEG) and then transmitted over to the EMS-user. Lastly, since all previous EMS systems allow only for brief, low-bandwidth interactions, we created *muscle-plotter*, which we describe in chapter 4. Unlike previous systems, muscle-plotter is capable of supporting sense-making activities, such as function plotting and math operations, by using a pen to persist the EMS output onto a piece of paper.

After providing the reader with an understanding of the prior art, we now return to our proposal to utilize EMS to create interactive systems. As we will see in the next chapter, the first property we uncovered in this dissertation was that EMS allows to substantially reduce the hardware footprint required to actuate the user's body and, thus, lends itself well to immersive applications (e.g., VR).

3 INCREASING IMMERSION USING EMS

As we established in Chapter 2, there has been a good amount of progress towards simulating the haptic qualities of *lightweight* virtual objects, such as contact with surfaces [86] or textures [28]. These haptic devices revolve around simulating the tactile qualities of the object by stimulating the user's skin; for instance, using inflatable pads at the user's fingertips [90] or vibro-tactile gloves [20].

Unfortunately, simulating the haptics qualities of *heavy* virtual objects, such as the weight of a piece of furniture or the resistance of a wall, has proven substantially more challenging. Even if one simulates the tactile aspects of such objects, the illusion fails as soon as users try to *push* through the virtual object, because their proprioception informs them about the lack of resistance [118].

Currently, the only haptic devices capable of rendering these large forces are motor-based haptic devices. As we have discussed previously, these devices can be stationary, such as the SPIDAR pulley system [107], or user-mounted, such as the SAM exoskeleton [93].

Regardless of their type, these devices are based on motors, which do not scale down gracefully while conserving the same output power. Thus, current haptic devices are bulky, heavy and require attaching to the limb they are actuating (so they can pull on it); for instance, a exoskeleton device that applies force to the user's hand will require the user to grasp a handle, which is then actuated by a pulley.

The objective of our research here is to explore EMS as an alternative to force feedback hardware and establish whether it is a suitable haptic device capable of increasing immersion.

3.1 Muscle-propelled Force Feedback: force feedback for mobile devices

Our starting point was to bring force feedback to mobile devices. In order to achieve this, we proposed using the user's muscle power as a replacement for motors by actuating the user's muscles using electrical muscle stimulation.

Figure 7 illustrates the use of our mobile force feedback prototype, in a mobile gaming scenario. The device is mounted on the back of a mobile phone, and the player has connected it to the palm flexor muscles of each of their forearms using two electrodes. The close-up reveals the game the user is interacting with. The game requires the user to steer an airplane through strong side winds by tilting the device. During he course of the game, wind turbines show up and create side-winds that derail the airplane. This effect is what our device renders as force-feedback.

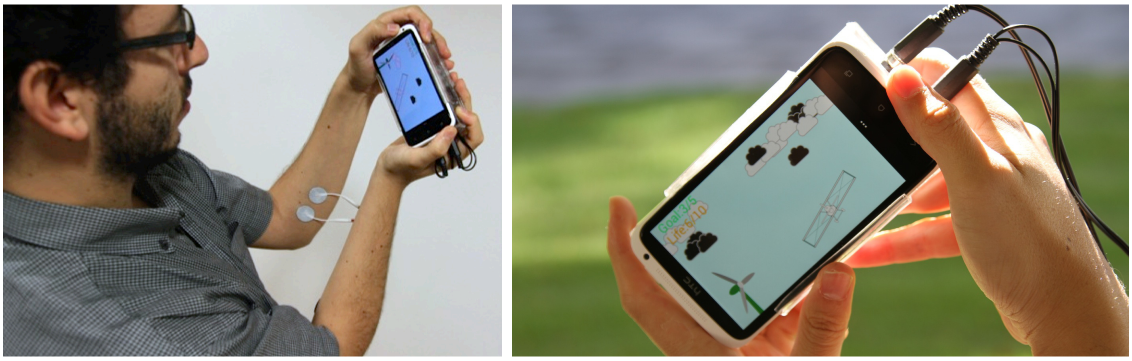

Figure 7: Our prototype electrically stimulates the user's arm muscles via the shown electrodes, causing the user to involuntarily tilt the device. As the user is countering this force, they perceive force feedback.

Figure 8 details how we produce the force feedback effect: the device renders the winds by trying to tilt itself against the user's will (Figure 8b). It achieves this by stimulating muscle tissue in the user's arm though the electrodes, triggering an involuntary contraction. This causes the user's arms to tilt sideways and thus the device to tilt. Since the airplane is controlled by tilt, the involuntary tilting threatens to derail the airplane. To stay on course, players counter the actuation using the force of their *other* arm (Figure 8c). As we find in Study 2, players perceive this as force feedback.

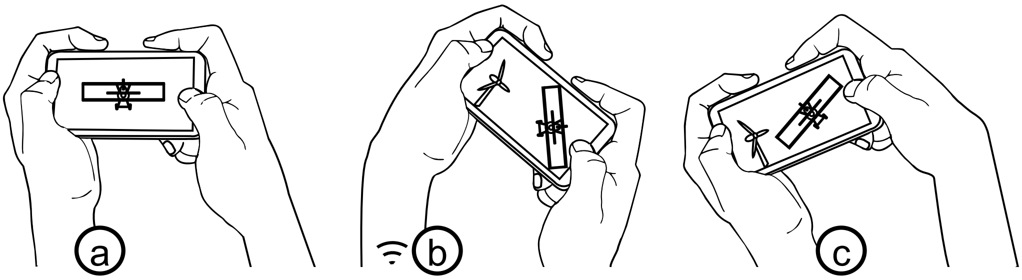

Figure 8: (a) As the user is playing (b) muscle-propelled force feedback kicks in, causing the user's left wrist to tilt the device. (c) The user responds by countering the forces, steering the plane against the wind.

3.1.1 Device hardware

The device induces involuntary muscle contraction by generating a biphasic waveform with a frequency of 25 Hz and a pulse width of 290 µs. Figure 9 shows a close-up of the hardware that produces this signal.

Our prototype measures 133 mm × 70 mm × 20 mm and weighs 163 g and is comprised of an *arduino nano* microcontroller, which communicates via USB or Bluetooth with its host device (*HTC One X* phone). The EMS generator is coupled to a medically compliant amplifier that outputs a maximum of 50V/100 mA over a 500 Ω load.

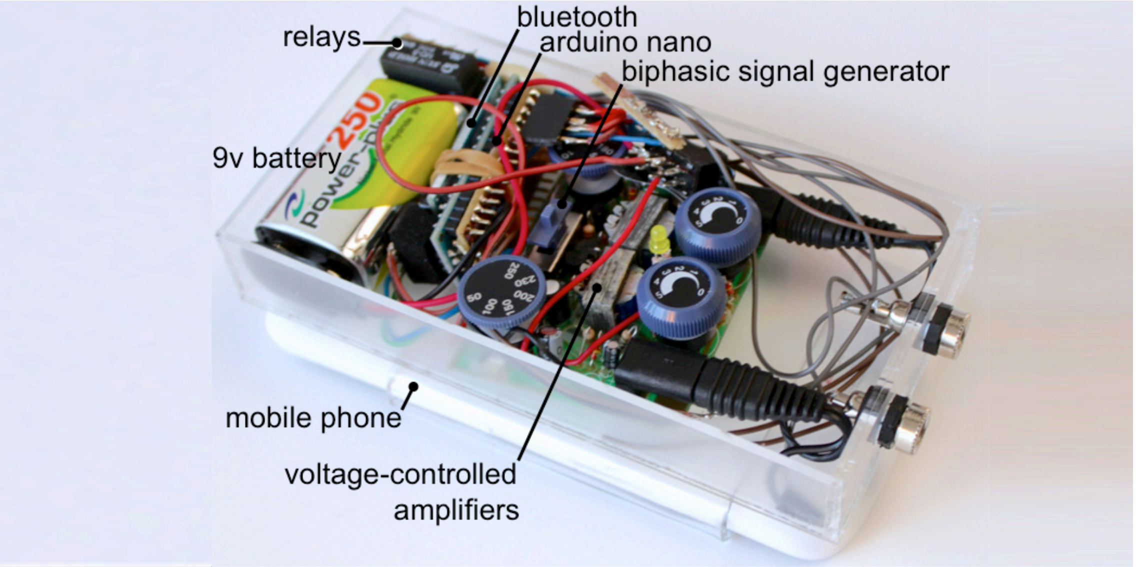

Figure 9: The backside of our prototype reveals its hardware design.

3.1.2 Contributions, benefits, and limitations

Our main contribution is the concept of creating mobile force feedback using computer-controlled muscle stimulation.

Our approach achieves miniaturization by (1) eliminating mechanical actuators, such as motors, and (2) substantially reducing battery size, as it is two orders of magnitude more energy-efficient to actuate a muscle (which receives its energy from the human body) than to drive a motor. In two simple user studies detailed below, we verified that our prototype created sufficient force and that its effect was indeed perceived as force feedback by the participants.

On the flipside, our device requires users to manually place electrodes, which requires expertise and takes some time. Future prototypes may overcome these limitations and achieve an even smaller form factor by using implanted electrodes [66].

3.1.3 Study 1: measuring EMS output force

To determine whether our approach delivers sufficient force, we evaluated the force of the muscular output induced by our prototype.

3.1.3.1 Participants

We recruited 10 right-handed participants (two female), between 24 and 50 years old (M=31.2 years old, SD=9 years), which had no prior experience with EMS. They received a small compensation for their time.

3.1.3.2 Apparatus and procedure

Our experimental apparatus, shown in Figure 10, actuated participant's wrist via pre-gelled electrodes on the palm flexor muscles (*flexor carpi radialis* and partially the *flexor digitorium superficialis*) and measured the resulting force using a digital spring-scale. Prior to the task, we calibrated an intensity range for each participant: minimum intensity with visible contraction up to the maximum intensity without causing any pain.

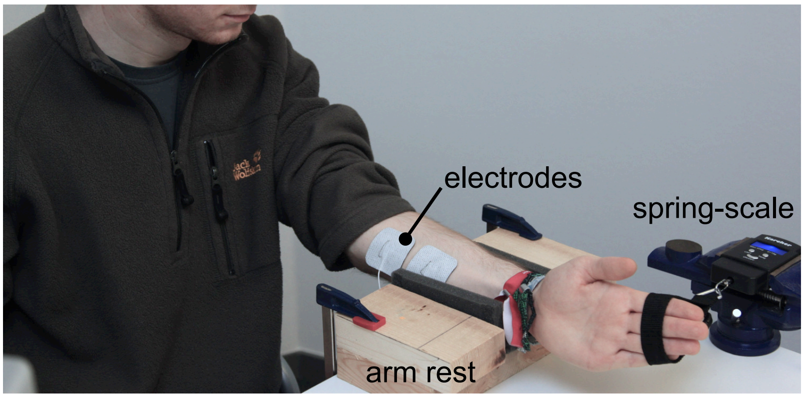

Figure 10: Apparatus for measuring the force an EMS-induced contraction.

3.1.3.3 Task & experimental design

For each trial, participants were subjected to a stimulation pattern and we measured the resulting force they produced. There were six stimulation pattern intensities (linearly interpolated between the minimum and maximum intensity values determined during calibration) and 11 durations (50, 100, 200…1000ms). Overall, each participant performed a total of 132 trials: 6 (intensities) × 11 (durations) × 2 (repetitions).

3.1.3.4 Results and discussion

Figure 11 shows the resulting forces for all users, showing that force grows with intensity level and is proportional to duration.

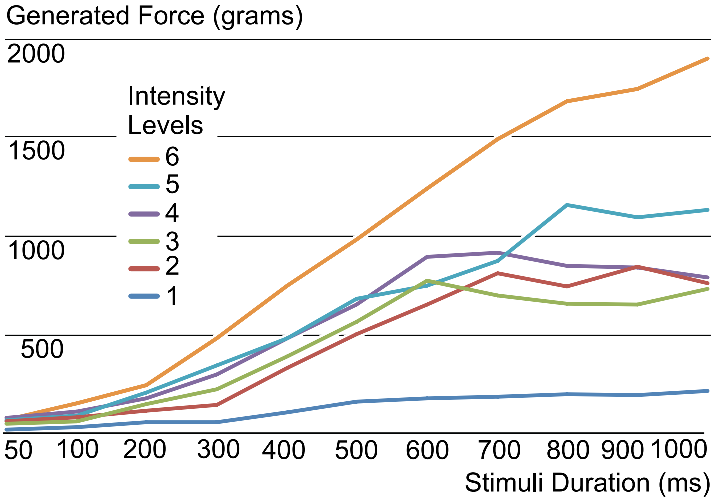

Figure 11: Average peak force (in grams) during palm flexion for each intensity.

At 1000ms of highest intensity stimulation participants produced an average of 1903g (18.7N). An increase in duration generally caused an increase of force for intensity levels 2 to 6. For the lowest intensity, however, force tapered off for stimuli longer than 500ms. For comparison, a Phantom force feedback device produces 3.3N [124]. Participants reached or exceeded this level for all intensities above 2 and stimulations longer than 400ms. This suggests that our prototype causes users to create sufficient force for typical force feedback applications.

3.1.4 Study 2: mobile force feedback gaming

In this study, we investigated how participants *perceive* the force feedback generated by our prototype. Participants played the game described above. We compared the experimental muscle-propelled *force feedback* condition with a *vibrotactile* baseline.

3.1.4.1 Participants

We recruited 10 participants (3 females) between 20 and 40 years old (M=27.4 years old, SD=5.4 years), from which none had partaken in Study 1.

3.1.4.2 Task

To win the game, participants had to keep the airplane on-screen while collecting white clouds and avoiding black clouds. Staying on screen required them to resist the winds that “pushed the airplane off-screen”. Participants steered the airplane left and right by tilting the device, while touching the upper and lower half of the screen using their thumbs allowed them to fly higher or lower.

3.1.4.3 Interfaces & experimental design

Prior to the study, participants were briefed about EMS and our device was calibrated to operate in such a way as to produce visible, yet pain-free contractions.

During the study (Figure 12), participants played the videogame with for at least 5 minutes per condition. In the *force feedback* condition, participants received feedback in the form of their screens tilting sideways under muscle actuation of both arms (i.e., both palm flexors). In the *vibrotactile* condition the direction of the wind was encoded using two different vibration patterns (participants were trained to correctly identify both patterns beforehand). The experiment used a within-subjects design and interface order was counterbalanced. After completing each condition participants filled in a questionnaire comprised of several 5-point Likert scale questions.

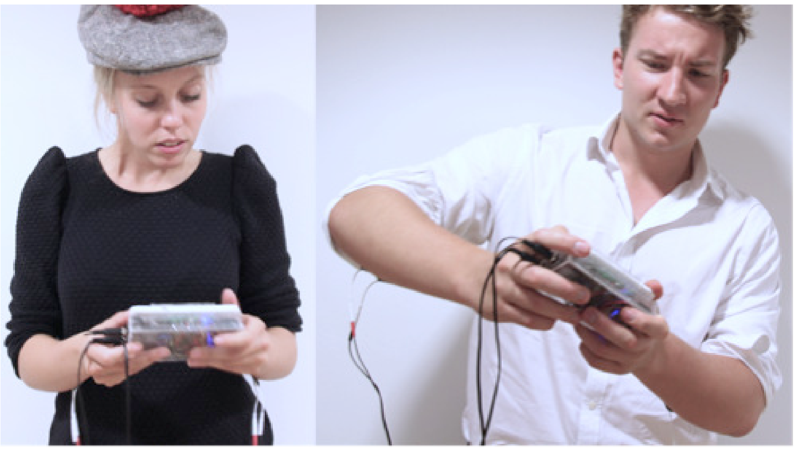

Figure 12: Participants from Study 2 experiencing the force feedback sensations delivered by our prototype (photos with participants' consent).

3.1.4.4 Questionnaire results & discussion

Participants rated the game as more enjoyable when playing with *force feedback* (Mdn=4.5 of 5, IQR=1), than *with vibrotactile* (Mdn=3 of 5, IQR=2), which a Wilcoxon Signed-rank test showed to be of statistical significance (Z=2.35, p =0.02).

Participants rated the wind forces in the game as harder-to-counter when *force feedback* was active (Mdn=4.0 of 5, IQR=2) than in the *vibrotactile* condition (Mdn=1 of 5, IQR=1), which a Wilcoxon Signed-rank test confirmed to be of statistical significance (Z=2.68, p =0.007). Perceiving the direction such winds in the game showed no statistical difference from *force feedback* (Mdn=5 of 5, IQR=1) to *vibrotactile* condition (Mdn= 3 of 5, IQR=2). All subjects expressed to prefer *force feedback* to *vibrotactile feedback*. Furthermore, participants' opinion suggested that *force feedback* contributed to a positive gaming experience (Mdn=4.5 of 5, IQR=1). Participants' reactions during the *force feedback* condition suggested excitement and included positive comments about an increased sense of realism and immersion. All participants stated that the muscle-propelled *force feedback* condition was pain-free (Mdn=1 of 5, IQR=0).

3.1.5 Conclusions on muscle-propelled force feedback

What we assessed with our first mobile prototype was that (1) using EMS we can create mobile force feedback devices that are smaller than motor-based solutions; and, (2) using a continuous EMS signal allowed us to generate strong, continuous and directional counter-forces. However, we also realized during piloting that these counter-forces generated by means of EMS felt somewhat artificial. This happens because while participants correctly feel the resulting counter-force, they also feel the current in their skin receptors (i.e., tingling) and their muscles contracting, which in turn reveals where the force actually comes from. Motivated by this shortcoming of EMS, we started to investigate how could we create more realistic sensations in a way that could perhaps mask the internal origin of the force.

3.2 Impacto: simulating impacts in virtual reality

In this project, our goal was to explore a realistic EMS-driven force sensation that, unlike in our previous project, did not reveal that the origin of the force was the user's own muscles. To this end, we focused on a specific category of haptic sensation, namely *impact*, i.e., the sensation of hitting or being hit by an object. Impact plays a key role in many sports such as boxing, fencing, football, etc.

Simulating impact is challenging though. Creating the impulse that is transferred when hit by a kilogram-scale object, such as a boxer's fist, requires getting a kilogram-scale object into motion and colliding it with the user. This requires a very heavy device. In addition, building up an impulse requires an anchor to push against, typically resulting in a tethered device, e.g., SPIDAR [107]. Both clash with the notion that today's virtual reality hardware is already wearable and wireless [69].

To achieve a realistic simulation of impact, we propose a different approach. The key idea is to decompose the impact stimulus into two sub-stimuli, each of which we can render effectively.

Our device, which we called *impacto*, was designed to render the haptic sensation of hitting or being hit. Figure 30 illustrates our approach, here at the example of a boxing simulation. The key idea that allows the small and light impacto device to simulate a strong hit is that it decomposes the stimulus. It renders the tactile aspect of being hit by tapping the skin using a solenoid; it adds impulse to the hit by thrusting the user's arm backwards using electrical muscle stimulation. Both technologies are small enough for wearable use.

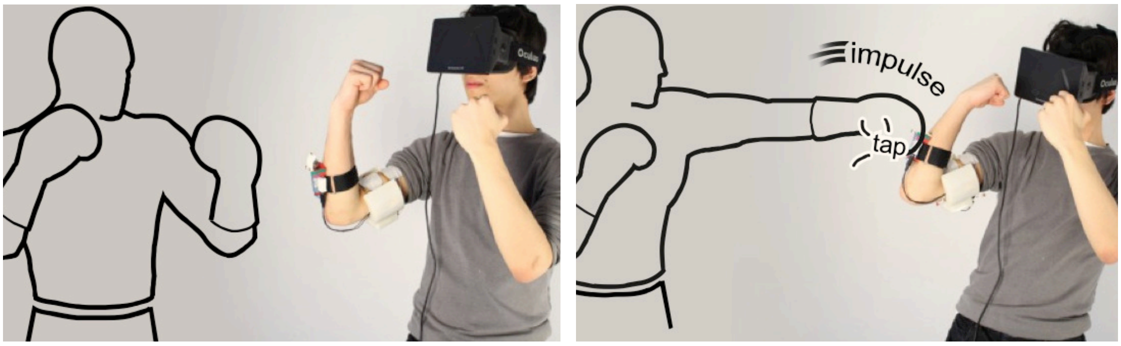

Figure 13: Impacto is designed to render the sensation of hitting and being hit.

Figure 14 shows the solenoid component in detail. To achieve a compact form factor, the solenoid is mounted parallel to the user's skin: a lever mechanism redirects its impact by 90 degrees, allowing it to hit the user's skin at a perpendicular angle. Furthermore, we provide a set of exchangeable 3D printed tips to refine the desired tactile experience, e.g., to simulate boxing without gloves we use a tip that resembles human knuckles (Figure 14c).

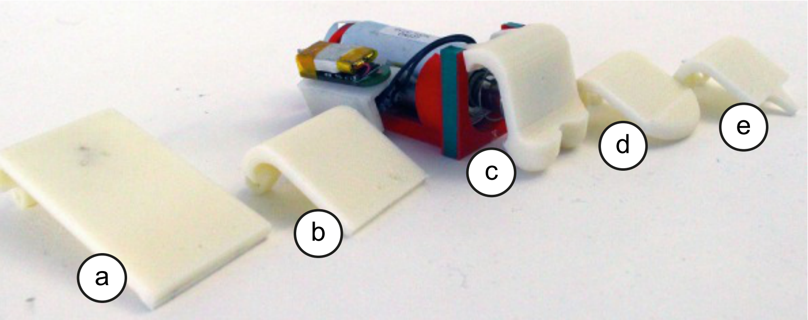

Figure 14: The solenoid component with a “knuckle” tip (c), which has a 90° lever to hit the skin orthogonally. The other four interchangeable tips are (a) generic surface, (b) small generic surface, (d) rounded, and (e) sharp.

Figure 15 shows the electrical muscle stimulation component. Its electrodes are mounted to the specific muscle that is able to render the impulse response that matches the solenoid; here, the solenoid is mounted to the outside of the arm, and therefore matches the impulse that would cause the arm to flex. Hence, we use the muscle that can flex the user’s arm, i.e., we attach the EMS component to the user’s biceps. When activated, the electrodes trigger an involuntary contraction of those muscles, simulating the transfer of impulse by thrusting the arm backwards.

Figure 15: Detail of the electrical muscle stimulation component. Here, it stimulates the user’s *biceps brachii* muscles causing an involuntary contraction that resembles force feedback.

Figure 16 shows the control unit that drives both solenoid and EMS components. We built impacto as a stand-alone wearable device, with all electronics embedded in a bracelet. The solenoid module features a Velcro closure, allowing the device to be strapped to the user’s upper arm, back of the hand, leg and so forth.

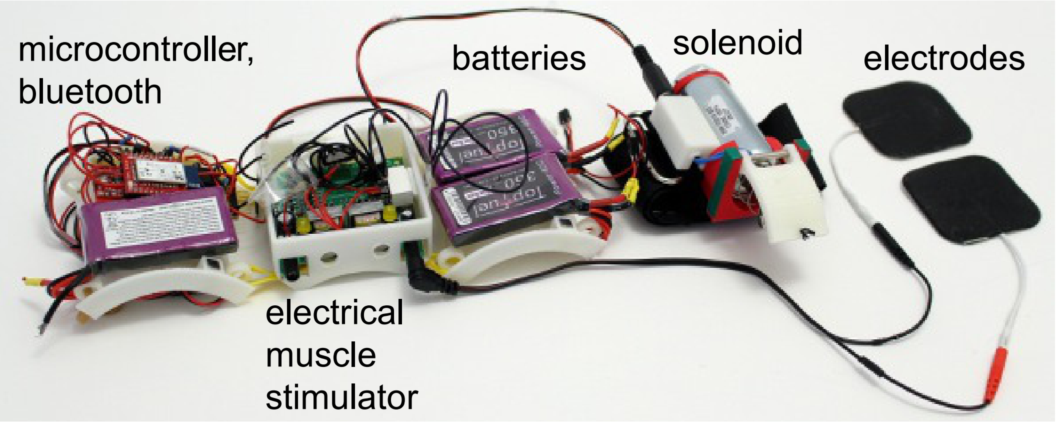

Figure 16: The impacto bracelet opened to reveal its contents.

3.2.1 Impacto's two components are mutually beneficial

Even though both technologies are small enough to allow for mobile or wearable use, it is their combination that creates a very strong sensation—in fact, stronger than either of the technologies by themselves (see impacto's "User Study"). However, solenoid and EMS play well together in more than one way:

1. **Impacto simulates an impulse**. The EMS component actually *moves* the arm. To create the required impulse, it creates a mechanical system between the limb and the user's torso. Given that the torso is comparably massive, there is very little effect on the torso and a strong effect on the limb.
2. **The physical response produced by EMS is strong,** despite the small form factor. It achieves this by leveraging the user's skeleton and muscles.
3. **Because of the EMS, the solenoid can be small, wearable.** Because the EMS is small but does the "heavy lifting", the task of the solenoid is limited to tapping the skin; this keeps the size of the solenoid down. With a small solenoid and EMS, we achieve a compelling simulation in a mobile/wearable form factor.
4. **A short EMS impulse does not elicit too much attention**. Unlike the continuous stimulation employed in our previous project, the stimulation used in impact is only present for 300 ms; during experimental piloting, we found this was enough to produce sufficient actuation force without eliciting much tingling sensation. Furthermore, when coupled with a simultaneous solenoid tap on the skin (in another location) the tingling sensation is further mitigated (perhaps masked).

3.2.2 Contributions, benefits, and limitations

Our main contribution is the concept of impact simulation, its decomposition into tactile and impulse components, and the implementation of these two components using solenoid and electrical muscle stimulation.

The main benefit of our approach is that it makes the simulation of a strong impact feasible in a small form factor. Our user study suggests that our approach generates a stronger sensation than either component in isolation. We demonstrate the use of our device in a series of virtual reality sport simulators.

On the flipside, simulating multiple impact locations requires multiple units, which places a natural limit on the spatial resolution of the simulation. Also, using a solenoid as a tactile feedback source adds inherent latency, which needs to be compensated for. Lastly, the use of EMS requires electrodes, which need to be manually placed by the user and calibrated prior to use.

3.2.3 Application examples

We have implemented three virtual reality sport simulators to demonstrate the potential use of impacto. All our examples use impacto for haptic feedback, an Oculus Rift for visuals and a Kinect for tracking. We now describe the applications from the perspective of what the user feels.

3.2.3.1 Hitting and being hit—Boxing

Boxing is a sport for which the notion of impact is crucial. Figure 17 shows a screenshot of the simple boxing simulator we created to experiment with impacto.

In this simulator, users can fight a virtual avatar by boxing. The avatar keeps its guard up and attacks periodically, and users must choose the right moment to unleash a successful attack. It takes ten successful hits to take down the avatar, which causes a new opponent to appear and the simulation continues.

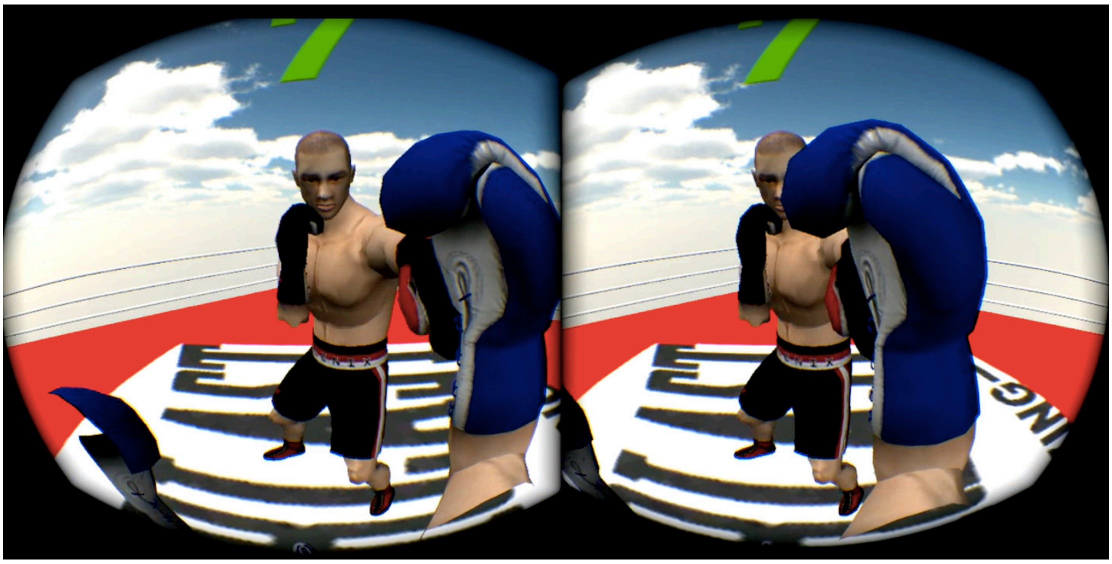

Figure 17: Stereo headset view from our simple boxing simulation. It allows users to attack the avatar and to block the avatar's attacks. Here, we see the avatar attacking the user's right arm.

Figure 18 illustrates how impacto adds a haptic component to the simulation: (a) The simulator provides haptic feedback when the user blocks the avatar, as discussed earlier. (b) The same impacto unit allows the user to hit the avatar using the part of the arm that wears the impacto unit, here the back of the arm.

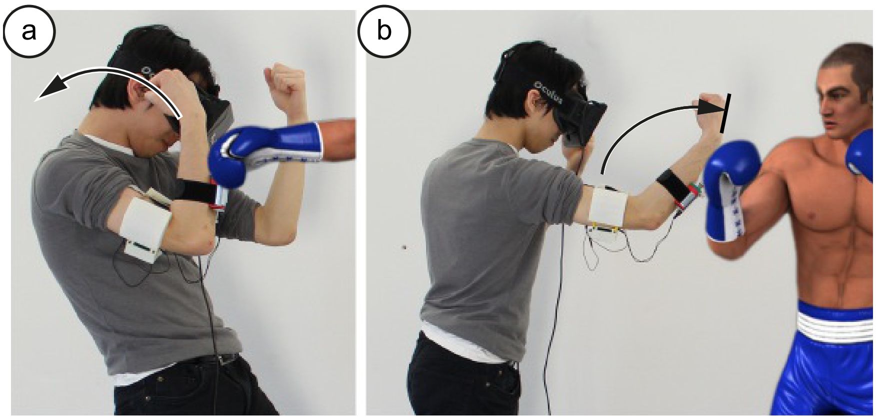

Figure 18: (a) Impacto allows users to feel the impact of blocking the avatar's hit by thrusting the user's arm backwards by operating the user's biceps. (b) The same impacto unit allows simulating the sensation of attacking.

Attacking using the back of the arm is an unusual (even illegal) attack in boxing. To allow the user to attack using other parts of the arm and/or to allow the avatar to attack additional targets on the user, we use additional impacto units. In the setup shown in Figure 19, we mounted a second solenoid component to simulate impact on the user's fist; this allows the user to attack using jabs and uppercuts. Since a knuckle hit leads to a similar impulse as the back of arm attack and block, we let both solenoid components share the EMS components on the user's biceps.

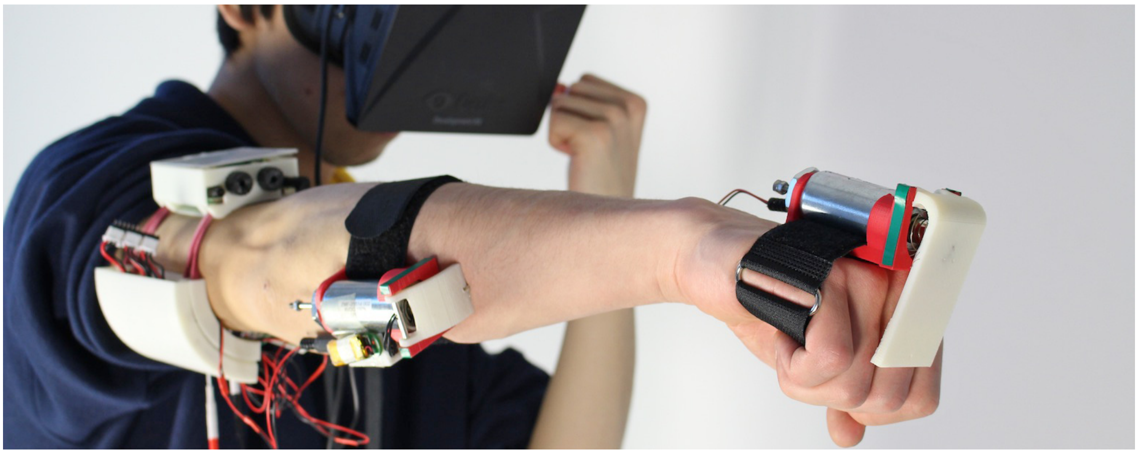

Figure 19: Additional solenoid unit mounted to the back of the hand featuring a surface tip allows the user to attack using jabs and uppercuts.

3.2.3.2 Applying impacto to other limbs—Soccer

Impacto units can be used on other limbs and muscles, such as the triceps, quads, etc. In Figure 20 we mounted a unit to the user's calves.

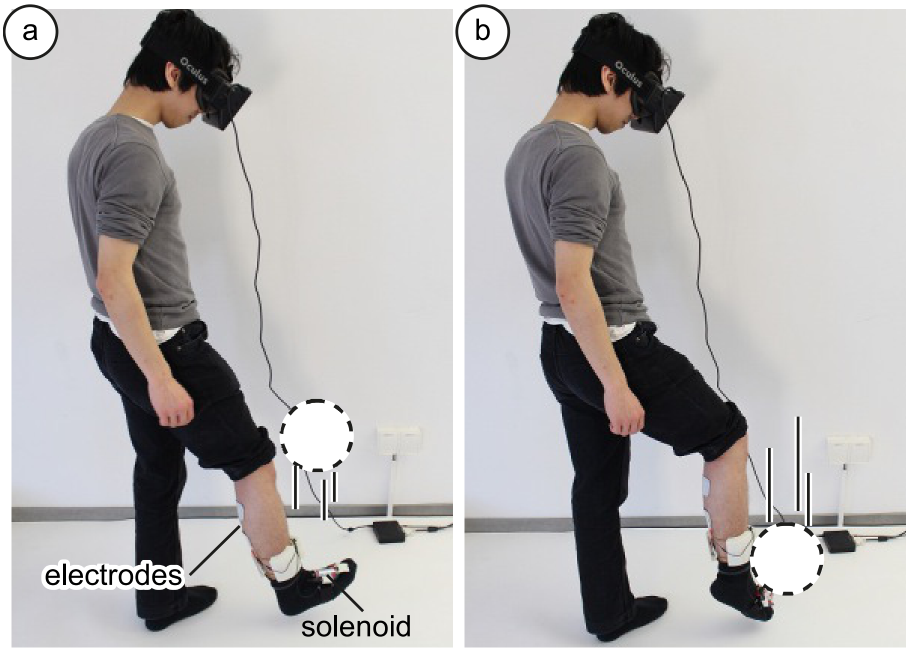

Figure 20: By wearing impacto on the leg and foot, the user experiences the impact of kicking a virtual football.

Figure 21 shows the simple simulator we implemented to illustrate this use case, which allows users to juggle a virtual soccer ball.

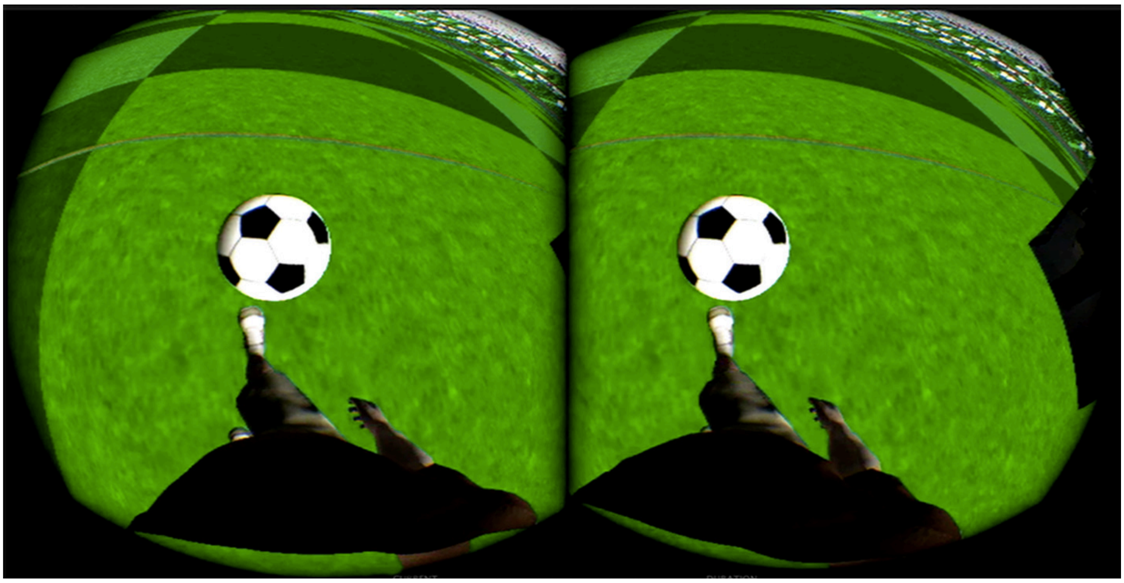

Figure 21: Stereo headset view from the football-juggling simulator. Here, the impacto unit renders the impact of a ball on the user's foot.

This setup points the solenoid component at the user's instep (top of the foot) and the EMS unit to the calf muscles (*gastrocnemius*), as depicted in Figure 22. We operate the unit so as to slightly push the foot backwards at the moment the ball hits the foot.

This football-juggling simulator uses the same Kinect setup as the boxing simulator. Additionally, to obtain the foot's tilt angle, we mounted a wireless accelerometer to the solenoid component.

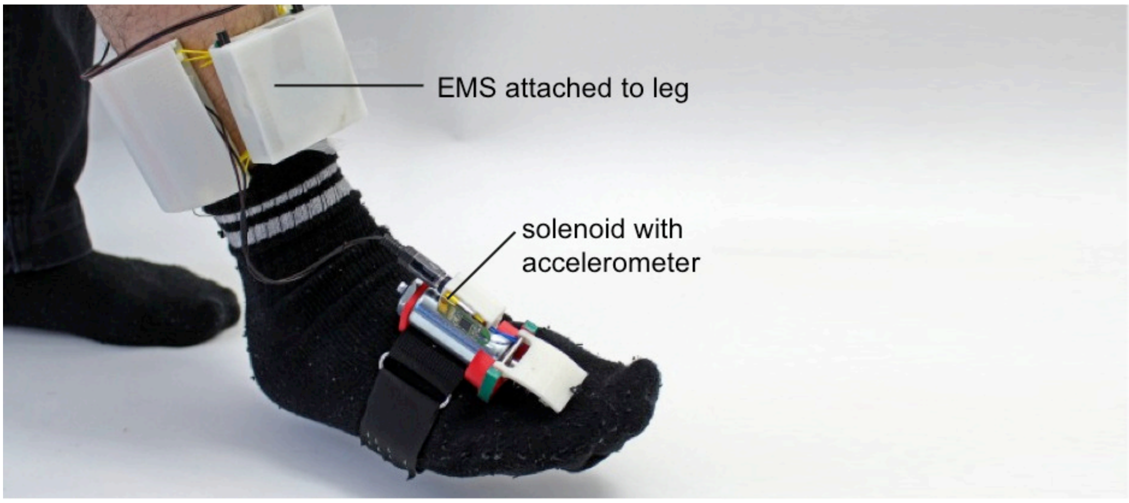

Figure 22: Close-up of on impacto unit mounted on the user's instep and calf. The wireless accelerometer on the solenoid senses the foot's tilt.

3.2.3.3 Combining impacto units—Thai Boxing

Users can experience impact sensations spread across multiple locations of their body by wearing multiple units. In Figure 23, we combined the setups from the boxing and the football-juggling simulator, resulting in a simple Thai boxing simulator.

Figure 23: A Thai Boxing experience with impacto.

3.2.3.4 Feeling Impact on Props—Baseball

The decomposition of impulse and tactile sensation transfers readily to hand-held props. Figure 24 illustrates this at the example of a simple baseball simulator. In the baseball simulator, by wearing an impacto unit, the user experiences the impact of an incoming baseball against the bat.

To enable the prop, here a stand-in for a baseball bat, we mount the solenoid onto the prop; the EMS unit, in contrast, stays with the user and stimulates the wrist extension muscle (*extensor digitorium)*.

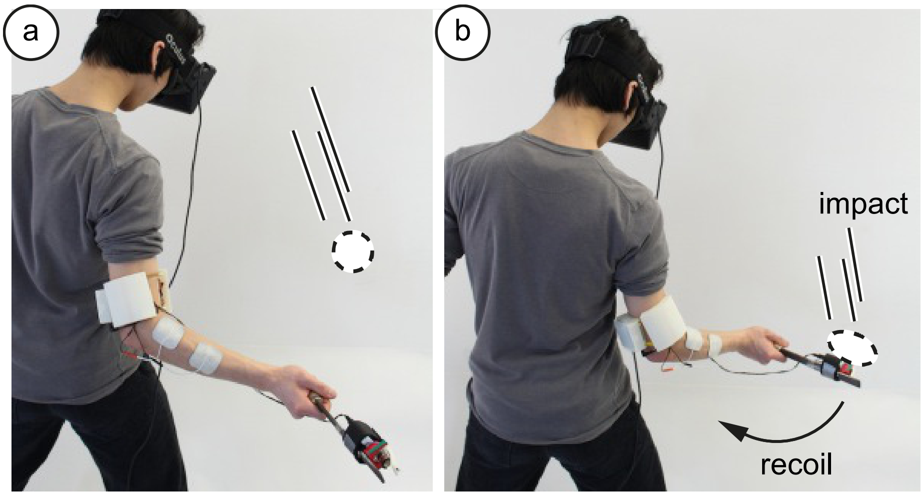

Figure 24: User feeling the impact of hitting a virtual baseball.

As illustrated by Figure 25, the same prop and electrode placement can power additional applications: by replacing the visuals in the virtual world and adjusting impacto's response, we can (a) reuse the same prop to simulate a (b) baseball bat, (c) a fencing weapon, or (d) a Ping-Pong paddle.

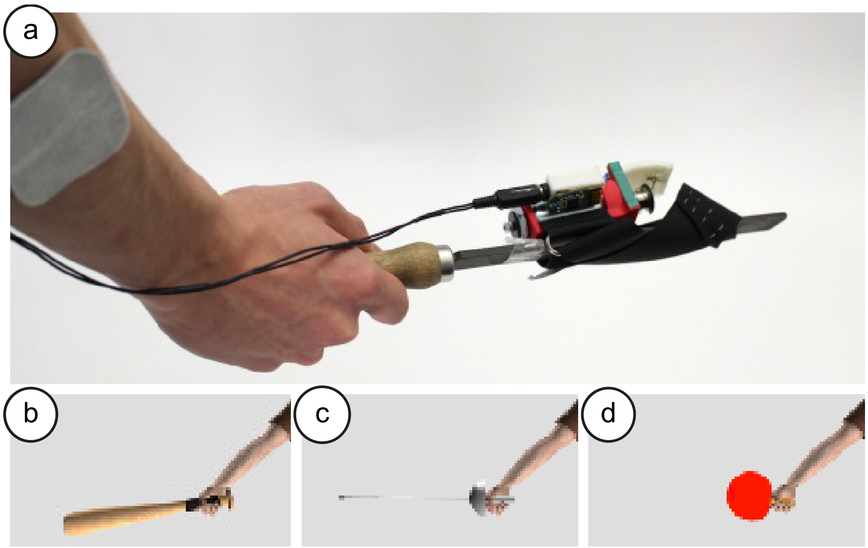

Figure 25: Achieving multiple haptic experiences via the same combination of impacto and a physical prop.

3.2.4 Implementation details

To help readers replicate our design, we now provide the necessary technical details of our hadware and software implementations.

3.2.4.1 Circuit design

Figure 26 shows the circuitry inside the impacto bracelet. The bracelet uses three 7.4 V LiPo cells in series for a total of 22.2 V and 1050 mAh to drive the solenoid in the boxing simulation; for simulations that involve weaker impacts, such as football, we used half the voltage. The estimated power consuption is: EMS (0.1 A), solenoid (0.5 A~0.7 A) and microntroller & bluetooth (0.2 A), allowing the unit to run for ~2000 hits.

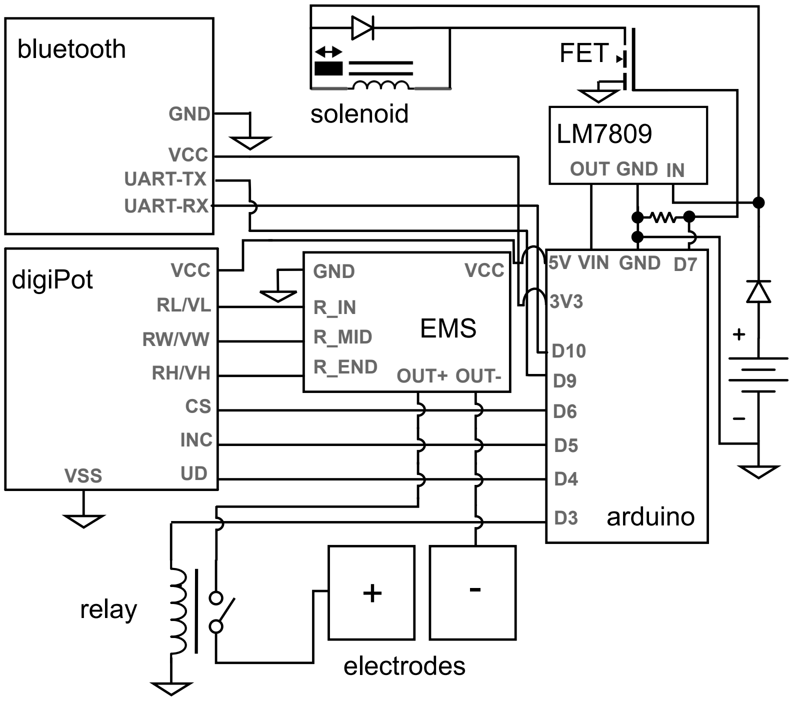

Figure 26: Schematic of impacto's circuitry.

The Arduino Pro Micro microcontroller (3.3 V, 8 Mhz) receives commands from the virtual reality applications via a bluetooth module (RN42XVP). The microcontroller and EMS unit (TrueTens V3) are powered through a 9 V voltage regulator (LM7809). The solenoid receives power directly from the battery (22.2 V). Optionally, the solenoid power can be regulated down to 20 V via another adjustable voltage regulator (LM317).

The unit can control EMS and solenoid intensity separately. One non-volatile digital potentiometer (X9C103) controls the intensity of the electrical muscle stimulation; the microcontroller controls it via a 3-wire protocol. One relay (HFD4/3) switches the EMS channel on/off in 3 ms. An N-Channel MOSFET (BUZ11) controls the intensity of the solenoid. It is sensitive enough to trigger at the low current output from the ATMEGA and can switch 30 A, which lies comfortably below the drain of our solenoid. The solenoid is bridged with a N4007 flyback diode to prevent the microcontroller from resetting due to the electromotive force that builds up when the solenoid is switched off. Modules that feature two solenoid outputs can switch between them using an additional relay (ommited from schematic).

The 3-axis wireless accelerometer in the football juggling application is an Axivity WAX3; it sends data wirelessly to the computer running the simulations. We pass the accelerometer data to the applications using the Axivity Wax library [8] via serial communications port. The accelerometer's internal battery allows the device to run for 8 hours.

3.2.4.2 VR simulators and tracking

All our applications use a Kinect to track the user's skeleton and Unity3D to simulate the virtual environments.

The Unity3D system detects collisions using collider objects attached to the skeleton of the user as represented in the virtual world. When a collision is detected, the system sends a serial message over Bluetooth to the impacto unit attached to that limb (each unit has its own Bluetooth address). The message contains which EMS channel and solenoid to trigger as well as the desired intensity. Users experienced all applications through an Oculus V1 head mounted display.

The solenoid mechanics and wireless communication are inherently subject to 60 ms of lag. One way to make the system appear instantly responsive is to have Unity3D using colliders with bounding volumes 25% larger than the actual limb, causing the collider to trigger ~30 ms early, thereby compensating for the lag of the system. On the flipside, this technique does not work for targets spatially clustered together or if the user stops abruptly before the target, as it creates a false positive.

3.2.4.3 Measuring latency

We determined the device's lag using a series of measurements on the apparatus depicted in Figure 27. This apparatus drives impacto's solenoid, making it tap a load-cell (MSP6951-ND) that was sampled at 1 kHz by an ATMEGA328 microcontroller, to measure the time difference.

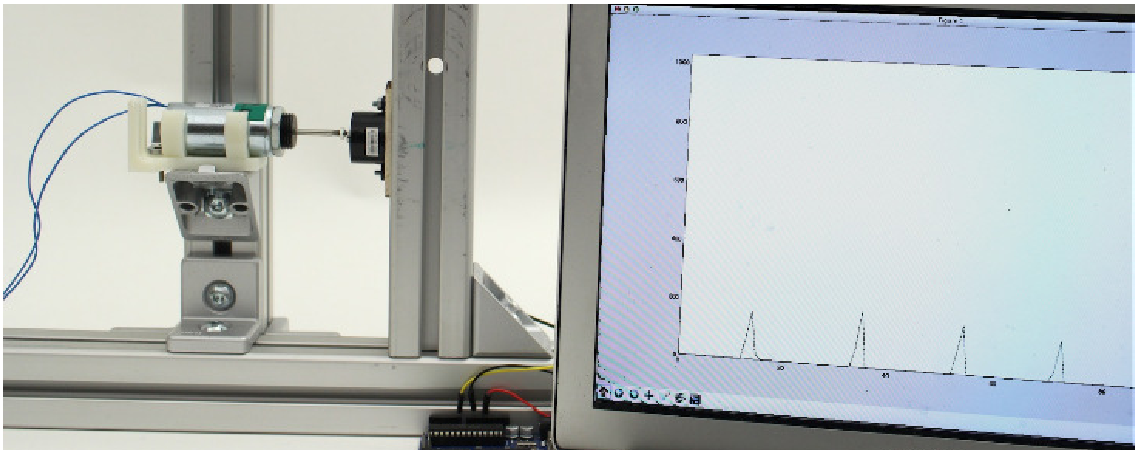

Figure 27: Apparatus for measuring force and latency.

As a baseline, we compare latency over Bluetooth to that of a direct serial connection, i.e., tethered over USB. Using a high-speed camera we measured 11 ms for the microcontroller to receive a single byte over USB and turn on an LED in response. The HFD4/3 relays take a maximum of 3ms to actuate (from datasheet). Using our apparatus, we measured that the solenoid takes 10~20 ms to extend fully and hit the load cell. Finally, our apparatus measured a latency of 50~60 ms for a tactile hit (Bluetooth + solenoid mechanics), which lies within the haptic threshold of 50-200 ms, as set by psychophysics research [130].

3.2.4.4 Measuring the energy transfer at the 90° lever

To validate the mechanical design, in particular the deflection lever, which pulls the tip using a fishing string, we conducted a series of force measurements.

The deflection lever redirects the solenoid's impulse by 90°, allowing the solenoid to be mounted parallel to the user's skin providing a much more compact form factor. We reused the apparatus shown in Figure 27 in two conditions, i.e., with and without the deflection lever.

Our measurements show that force exerted by the vertical hit (as in "User Study") is 26 N, while for the horizontally mounted solenoid, which hits through the 90° lever, we measured 21.1 N. Measurements are an average of 10 hits on the load-cell using the knuckle tip. These measurements clarify that both setups are comparable.

3.2.5 User study

To validate the core idea behind impacto, i.e., the idea of decomposing an impact's haptic feedback into a tactile component (*solenoid*) and an impulse component (*EMS*), we conducted a user study.

To do so, we immersed participants in a simplified study version of our boxing simulator in which they blocked punches unleashed by an avatar opponent. We varied the intensity of *solenoid* (no, low, high) and *EMS* (no, low, high) in a full-factorial design and asked participants to assess the realism of the punches. We hypothesized that the combination of both stimuli would lead to a more realistic experience.

3.2.5.1 Apparatus

Figure 28a shows our apparatus. Participants wore a head-mounted display (Oculus Rift V1). A single impacto unit was mounted to their right forearm, with the electrodes of the EMS component attached to the participant's *biceps brachii* muscle.

We used an earlier design of impacto, but, nevertheless, it used the same EMS component and produced similar output force conditions as the bracelet (see previous section). For the tactile sensation we used the knuckle tip.

To ensure a controlled experience we used a scripted version of our boxing simulation, in which a video avatar repeatedly punched the participant (Figure 28b) on the dorsal side of their right forearm.

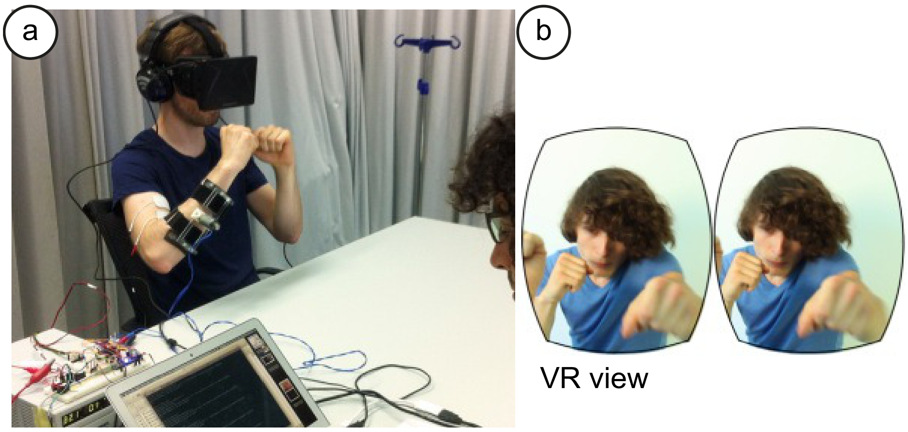

Figure 28: Our experimental setup to test impacto.

Participants were seated and held their arms in a guard position, so as to match the hands they saw in the video experience. Participants rested their elbows on the table between trials to reduce fatigue.

3.2.5.2 Interface conditions

There were nine interface conditions, i.e., the full-factorial design of *solenoid* intensity (*no, low, high*) and *EMS* intensity (*no, low, high*).

In the *high EMS* conditions, the EMS component was calibrated to perform a full biceps curl, i.e., a 45 degrees movement from the default guard pose. In the *low EMS* conditions, the EMS component was calibrated so as to create the weakest visible contraction of the participant's biceps. In the *no EMS* conditions, the EMS component was off.

During setup, we made sure that participants felt conformable with the setup and reached 45 degrees without any discomfort; this was the case for all participants.

In the *high solenoid* condition we overdrove the 12 V solenoid with 32 V for 200ms, resulting in a strong (~26 N) tap. In the *low solenoid* condition we operated the solenoid at its nominal voltage of 12 V for 200 ms resulting in a weaker (~13 N) tap. In the *no solenoid* condition, the solenoid remained off.

3.2.5.3 Task and procedure

For each trial, participants observed a 9 seconds video experience of being punched against their guard 3 times. This was accompanied by the respective haptic feedback created using the impacto unit.

Participants then rated the realism of the punches on a 7-point Likert scale (1 = artificial, did not feel like being punched, 7 = realistic, like being punched).

Each participant performed a total of 27 trials: 3 force feedback settings (*no EMS, low,* or *high EMS* strength setting) × 3 tactile feedback settings (*no solenoid, low*, or *high*) × 3 repetitions. This yields a 3 × 3 within-subjects design.

The EMS calibration procedure took about 4 minutes during which the biceps contraction was repeated ten times to ensure that a similar contraction was found.

3.2.5.4 Participants

We recruited 12 participants (3 female), between 22 and 35 years old ($M = 26.9$ years) from a nearby organization. We excluded a thirteenth participant from the analysis who had stated that he/she had started with too high ratings, thereby producing a ceiling effect. One of the participants had boxing experience (sparring) and another was trained in martial arts. Two participants had never experienced a VR headset before and only one had experienced EMS before (in physiotherapy). With consent of the participants we videotaped the study sessions.

3.2.5.5 Results

Figure 29 shows the resulting data, i.e. participants' assessment of the realism of the punches as a result of the different haptic feedback conditions. We analyzed the data using a 3 *(EMS)* × 3 *(solenoid)* × 3 *(repetition)* repeated measures ANOVA ($\alpha = .05$) as suggested by [113]. Since we found no learning effect as there was no main effect of repetition ($F_{2,25} = 0.225$, $p = .800$), we used all three repetitions as data.

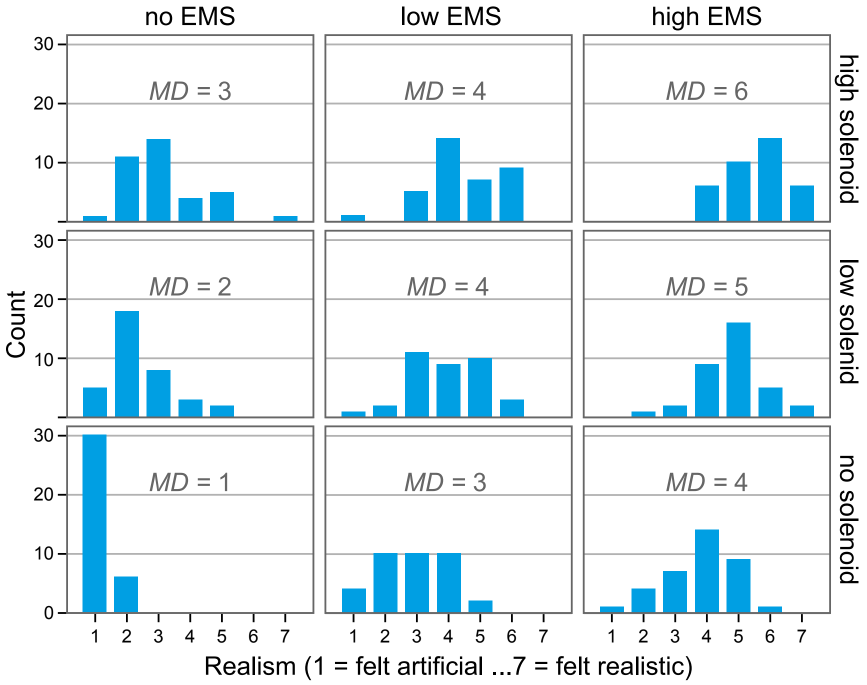

Figure 29: Realism ratings in dependence of force feedback (*EMS*) and tactile feedback (*solenoid*) conditions.

As expected, we found main effects for force feedback (*EMS*, $F_{2,14} = 89.726$, $p = .000$) and tactile feedback (*solenoid*, $F_{2,14} = 56.840$, $p = .000$, Greenhouse-Geisser corrected for sphericity), i.e., *higher solenoid* intensity and *higher EMS* intensity both led to more realism. We did not find any interaction effect of *EMS* * *solenoid* ($F_{2,14} = 1.524$, $p = .210$). Post hoc pair-wise comparisons using the test (Bonferroni corrected) confirmed the statistical differences across intensity levels for both, *EMS* (all pairwise comparisons, $p < .001$) and *solenoid* (all pairwise comparisons, $p < .001$).

3.2.5.6 Participants' feedback after the experience

After finishing all trials we interviewed participants about their experience: Seven participants stated that they found the experience "immersive". Referring to the first time he/she had felt the combined effect P4 stated "it got immersive after a while, when I felt a stronger hit for the first time". P7 said "the first time I felt it, it was surprising, felt like a realistic force". P4 also added "I felt I needed to protect myself from the hits, it got real for me". P10 went further and stated: "this seems to really help VR, it is the most realistic VR experience I've ever had". P3, who was acquainted with boxing/sparring, stated "I know the feeling [impact] from sparing and this was really cool, could be even stronger [the solenoid hit]" and added "it is really impressive that this actually moves my arm".

All participants stated that they liked the combined effect better than the individual effects, as suggested by their earlier assessments of "realism"; P5 explained "the stimulation does not feel like a hit, but the combination really feels real because I suppose if you get hit your muscle moves back after the skin is hit". P8 said "I clearly felt that a hit [solenoid] and response [EMS] made it much more real". P7: "The solenoid feels like a punch and so it is more important, but then only with the EMS it felt real". Similarly, P9, who had 10 years of martial arts experience, said "solenoid is more important because it is like getting hit, but I prefer when both are on." P10 said "The EMS helps, but the primary thing is that it touched me." P12: "if you have solenoid, then the EMS really helps me to feel [that it is] real". P2 said "I was skeptical of the EMS during the calibration, but when I saw it in combination with the VR video and the solenoid, it was impressive".

Four participants stated that without the solenoid the experience feels unrealistic, such as "without the solenoid it was hard to understand when [the virtual boxer] hit me" (P3).

While we hypothesized that a shorter pulse would minimize the tingling effect, three participants that pointed out that the residual EMS tingling had slightly affected their sense of realism "I felt it vibrating, so that is a bit different from pure movement" (P4).

When asked “what is missing for a fully realistic experience” participants answered: “resolution of the headset” (P10, P8), “remove the tingling caused by the EMS” (P12, P13), “it should also actuate my shoulder” (P4), and “the tactile part should be a larger surface, like a fist model” (P11, P9).

3.2.5.7 Discussion of study findings and study limitations

Our study found main effects on both EMS and solenoid, suggesting that increasing the intensity of either of the haptic effects increases the perceived realism.

Secondly, the highest score was achieved by combining both stimuli, supporting our hypothesis. Participants’ comments further support that hypothesis in that *all* participants stated that the combined effect had felt more realistic than either of the individual effects. While our pilot explorations suggest that a short EMS impulse is less noticeable, our study is limited in that it did not study the particular effect of different lengths of EMS stimulations.

3.2.6 Conclusions on impacto

With impacto we were able to create a more compelling force sensation by (1) utilizing a very short EMS window (300 ms instead of continuously stimulating the muscle) in combination with (2) adding a tactile stimuli that “justifies” the EMS-induced motion, i.e., the user’s arm moves because it was “hit” by the object currently touching the arm. As our user study assessed, this combination seems to provide users with a sufficiently believable explanation and does resemble a punch in ordinary life. Lastly, our findings hint that to some degree we can think of the tactile impulse as a distractor, which masks the user’s attention away from the tingling sensation of the EMS.

However, the sensations this prototype is able to deliver are limited to brief impact moments. We now turn our attention to our initial question: *imagine stretching your arm in virtual reality and pushing against a virtual wall.*

We will see how the key principle we found through impacto, i.e., add a “justification” to mask out or explain the EMS as the origin of the force, will assist us in developing a system that emulates the sensations that arise from interaction with large masses, such as walls and heavy objects.

3.3 Providing haptics to walls & heavy objects

As we surveyed in Chapter 2, the two traditional approaches to simulating the forces arising from manipulating objects in VR are: (1) to tether the user's hands (e.g., *SPIDAR* [107], or (2) wear an exoskeleton [93]). However, as we discussed both these devices limit the user's range of motion, as they are heavy and extremely cumbersome.

Thus, our goal was to render heavy objects in VR in a truly wearable form factor. Our main idea is to prevent the user's hands from penetrating virtual objects by means of electrical muscle stimulation (EMS). Figure 30a shows an example: as the shown user lifts a virtual cube, our system lets him feel the weight and resistance of the cube. The heavier the cube and the harder he presses the cube, the stronger a counterforce the system generates. Figure 30b illustrates how our system implements the physicality of the cube, i.e., by actuating the user's *opposing* muscles with EMS.

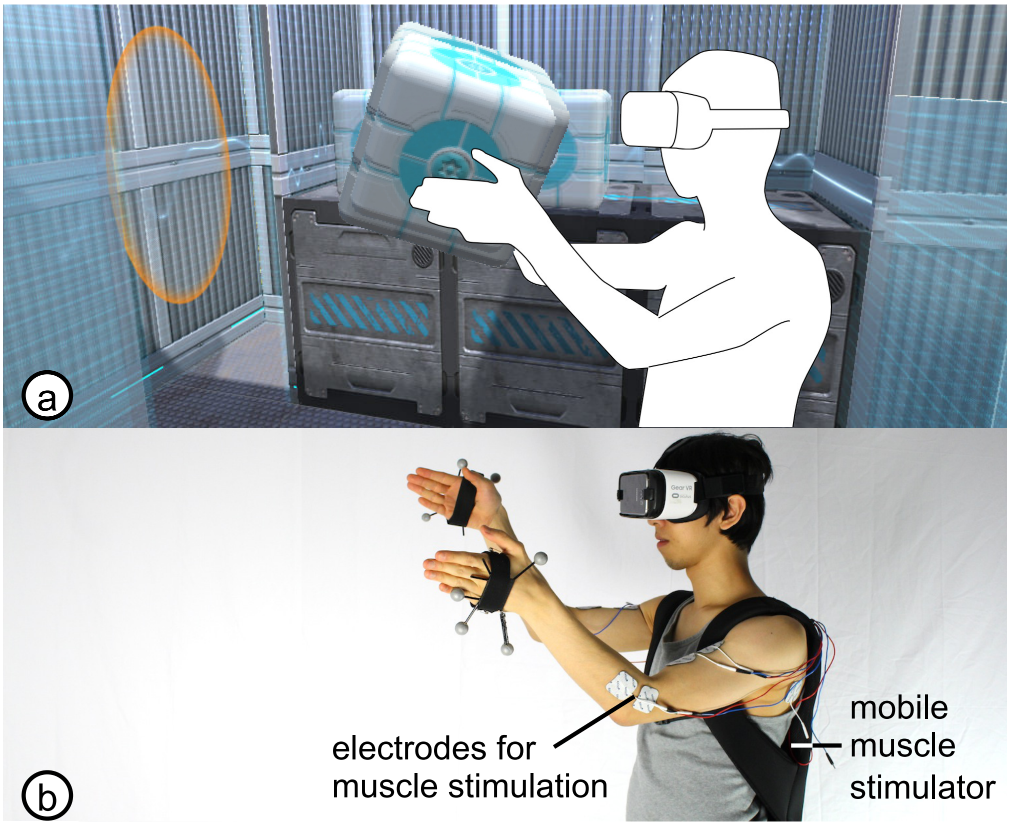

Figure 30: (a) As this user lifts a virtual cube, our system lets the user feel the weight and resistance of the cube. (b) Our system implements this by actuating the user's *opposing* muscles using electrical muscle stimulation.

Figure 31 illustrates the idea in more detail. (a) When the user grabs the virtual cube, the user expects its weight to create tension in their *biceps* and the cube's stiffness to create a tension in their *pectoralis*. (b) In order to create this sensation, our system actuates the respective *opposite* muscles. So, to put a load onto the biceps, it actuates the *triceps*; and, in order to load the *pectoralis*, it actuates the *shoulder muscle*. This creates the desired tension in biceps and pectoralis, thereby creating the desired experience.

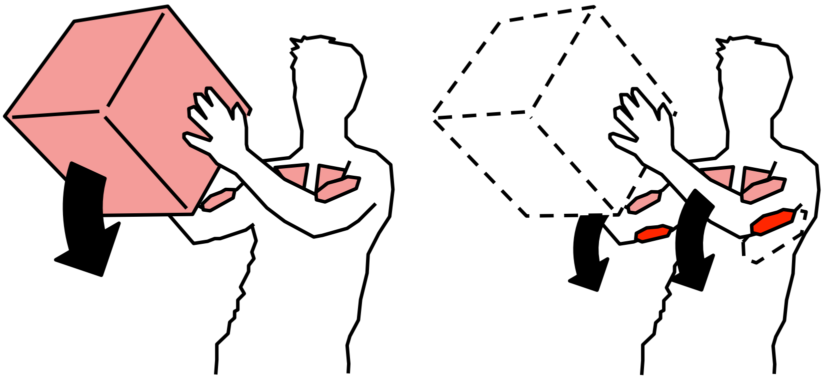

Figure 31: (a) When a user picks up a physical cube, its weight causes tension in the user's biceps. (b) Our system creates this tension by instead actuating the opposing muscles, here the user's triceps and shoulders.

As illustrated by Figure 32, our system stimulates up to four different muscle groups. Through combinations of these muscle groups, it simulates a range of effects: when pushing a button mounted to a vertical surface, for example, the system actuates biceps and wrist. In the "Example widgets" section, we detail how this enables the simulation of a wide range of objects including walls, shelves, buttons, projectiles, etc.

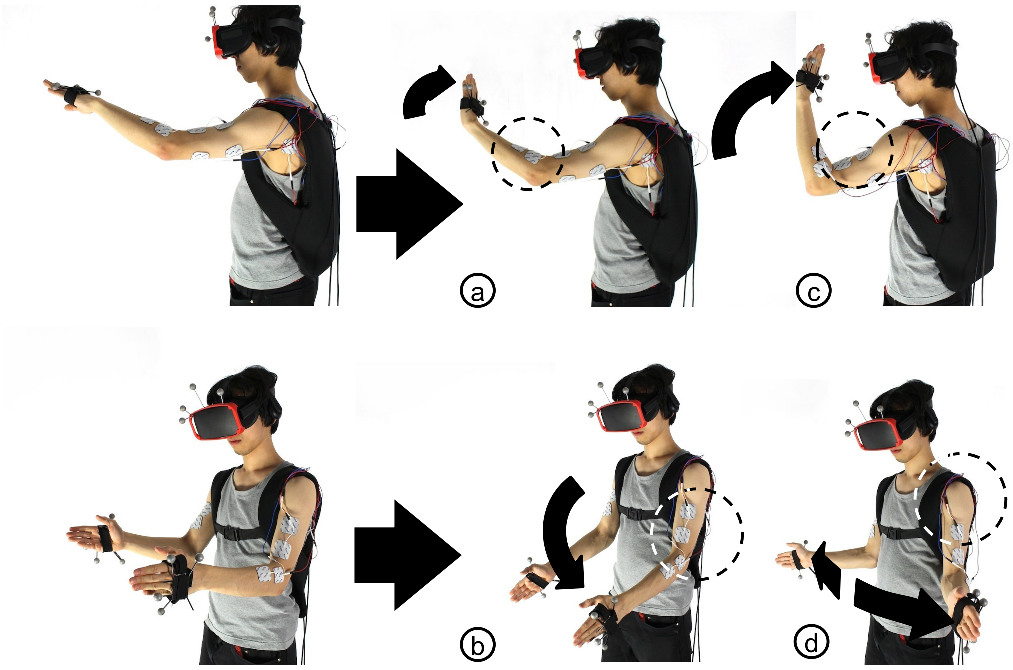

Figure 32: We use up to 8 electrode pairs, actuating (a) wrist, (b) biceps, (c) triceps, and (d) shoulders.

Our system can be worn in a small backpack, as shown in Figure 32. The backpack contains a medical compliant 8-channel muscle stimulator (see "Implementation details" section), which we control via USB from within our VR simulators. We use the system in the context of a typical VR setup consisting of a head-worn display (using a Samsung/Oculus GearVR) and a motion capture system (based on eight OptiTrack 17W cameras).

3.3.1 Contributions, benefits, and limitations

Our main contribution is the concept of providing haptics to walls and other heavy objects by means of electrical muscle stimulation. We achieve this in a wearable device, suitable for real-walking virtual reality environments.

Limitations include the need to wear EMS equipment and the design space, which works best for soft and repulsive objects, rather than truly rigid objects. Furthermore, we designed our haptic effects based on eight muscles from both shoulders and arms. These eight muscles alone were sufficient to create a plethora of haptic VR objects and obstacles. While adding more channels would possibly result in more complex haptic effects, it would also require attaching more electrodes and EMS hardware.

3.3.2 Design

Based on the (general) concept of using EMS to bring force feedback to VR we can now design the user's experience.

Two dimensions have substantial impact on the experience: (1) The intensity pattern we use to actuate the user's muscles and (2) the visuals and sound we present during this haptic event. It turns out that both are crucial, in that they determine what physical event users will associate with the haptic sensation. These two aspects are also crucial for making the experience convincing.

Ideally, a design should fulfill four criteria, presented in order of decreasing importance:
1. **Believable**: users have the idea that the virtual object is causing the experience.
2. **Impermeable**: prevent users from passing through the object.
3. **Consistent**: visual and haptic sensation should match.
4. **Familiar**: the experience should ideally resemble objects from the real world.

3.3.2.1 The hard object design does not work

Figure 33 illustrates the naïve approach to rendering objects using EMS: (a) From the moment the user's fingertips reach the virtual wall, we actuate the user's hand just strongly enough to prevent it from passing through. We achieve this with a current essentially proportional to the user's force (further details in Implementation).

When we built this version, the results *looked* great. The design prevented the user's hand from passing through the object and thus bystanders observing the scene would typically conclude that the illusion was "working". However, during piloting it became clear that this design did *not* work. Since the EMS actuation was as long and as strong as the user kept pushing, the EMS signal (a tingling in the respective muscles) could become arbitrarily strong. This would draw the user's attention to the EMS-actuated muscles. These, however, were pointed in the wrong direction, i.e., they were pulling, when the sensation was supposed to be about pushing. One participant in our pilot said this design felt "like a magnet pulling the hand backwards".

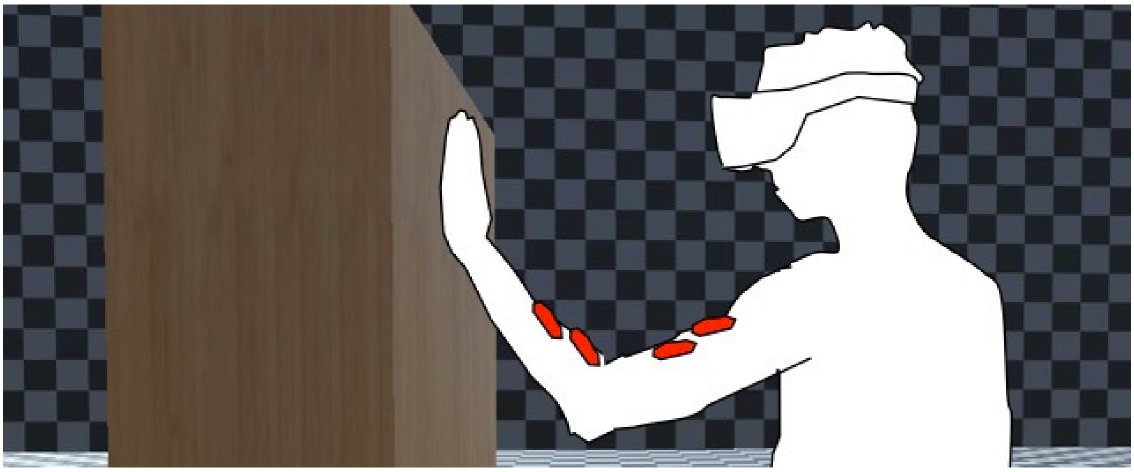

Figure 33: Implementing rigid walls requires stimulating muscles with strong impulses over long periods. This draws undesired attention to the electrical stimulation.

While this design was reasonably *impermeable*, *consistent,* and definitely *familiar*, the strong EMS signal made this design fail with respect to our primary objective: it was not *believable*. We therefore created two alternative designs with the objective of increasing *believability*.

In order to avoid the *long and strong* EMS signal that had made our actuation obvious, we created one design based on a *weaker* signal and another based on a *shorter* signal.

3.3.2.2 The soft object design

We created our first alternative design by applying a cut-off to EMS intensity. We picked a reasonably low cut-off, allowing users to penetrate objects by about 10 cm. This resulted in a design that produced the impression of *soft* objects.

Based on this general concept, we explored various visuals, including the soft surface material shown in Figure 30, which is designed to suggest an increasingly solid inside under a soft, permeable surface. This general design became the basis for most of our object designs.

Figure 34 shows the same concept wrapped in visuals suggesting a *magnetic field*, suggesting a magnetic force that carefully pushes the user's hand backwards. In some versions of this design, we attached a block of metal to the back of users' hands to suggest that the magnetic field would apply there in order to affect the hand.

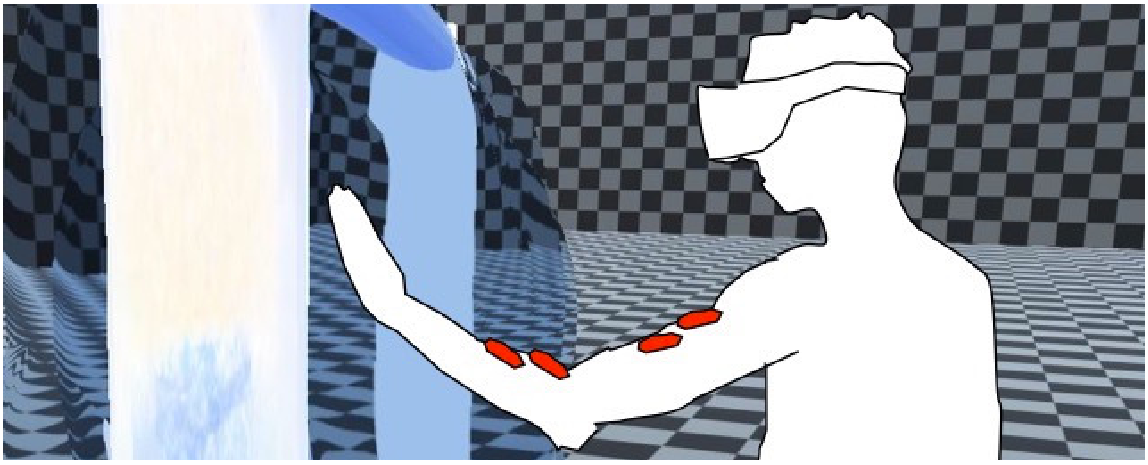

Figure 34: The *magnetic* visuals allow the user's hand to penetrate the objects.

3.3.2.3 The repulsion object design

We created our second alternative design by reducing the *duration* of our EMS signal. This resulted in what we call *repulsion* objects. This design uses a brief EMS pulse (of 200-300 ms, using the user's calibrated maximum intensity) where the EMS propels the user's hand backwards, removing it from the virtual object it is trying to touch. We achieve this with an EMS pulse of still reasonably low intensity, which, like all other EMS signals in our system, is pain free at all times (for EMS pulses of similar intensity see impacto). Again, we explored various visuals with this haptic design in order to help users rationalize what happens when they touch the object. Figure 35 shows what we call *electro* visuals.

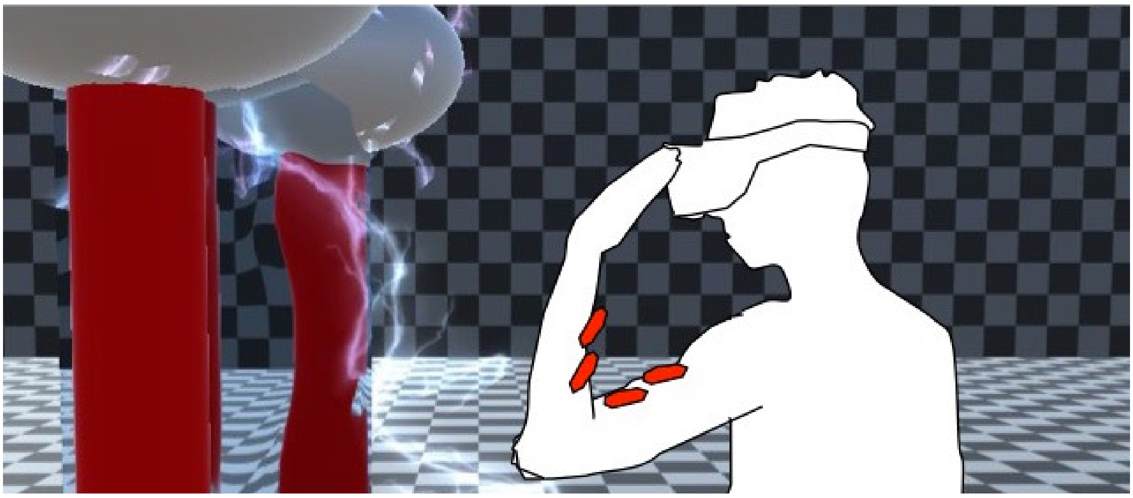

Figure 35: The *electro* visuals.

This design complements the EMS pulse with a strong white flash that turns the screen white for 100 ms and then fades it back in 100 ms; at the same time, users hear a loud electrical “bang”. To reinforce the effect further, we artificially enlarge the visual appearance of the user’s hand movement, making it appear as if it was thrust backwards even further. In some cases, we complemented our EMS pulse with a strong vibration motor mounted to back of users’ hands (an eccentric 5V DC motor operated at 12 volts) to suggest an electric flash hitting the user’s hand.

This gave us two functional designs, i.e., one to represent soft surfaces, as well as the repulsion design, which we would later use as a stand-in for hard surfaces.

3.3.3 User study 1: validating designs

We conducted a user study in order to (1) validate our core idea of using EMS as a means for adding haptics to heavy objects in virtual reality and (2) to validate the qualities of our *soft* and our *repulsion* object design.

We immersed participants in a simple virtual world that contained nothing but five walls, each featuring a different haptic design. Participants touched all five of them and rated their qualities. We hypothesized that the *soft* and the *repulsion* design would perform best.

3.3.3.1 Interface conditions (i.e., five virtual wall designs)

Figure 36 shows the five “walls” arranged in a pentagon with the participant inside. Each “wall” implemented one of five interface conditions:

1. The *soft* wall used the *magnetic* visuals from Figure 34.
2. The *repulsion* wall used the *electro* visuals from Figure 35.

We also included three additional conditions featuring more conventional visual explanations of a “hard” wall, all of which employed the visuals of a solid wooden wall as depicted in Figure 33.

3. The *soft wood* wall was identical to the *soft* wall, in terms of the EMS feedback, yet depicted a solid wooden wall. We used this condition to validate whether the visual design of the *soft* wall would add to the experience.
4. The *soft vibro wood* wall was identical to the *soft wood* wall, but also actuated the vibrotactile actuator on the back of participants’ wrists. We used this condition to test whether vibro tactile would add to the experience.
5. The *vibro only* wall, finally, actuated *only* the vibrotactile actuator on the back of participants’ wrists, but did *not* provide any EMS feedback. This condition served as baseline, as vibrotactile is the most common conventional approach to rendering haptic feedback in virtual reality (as described in Chapter 2).

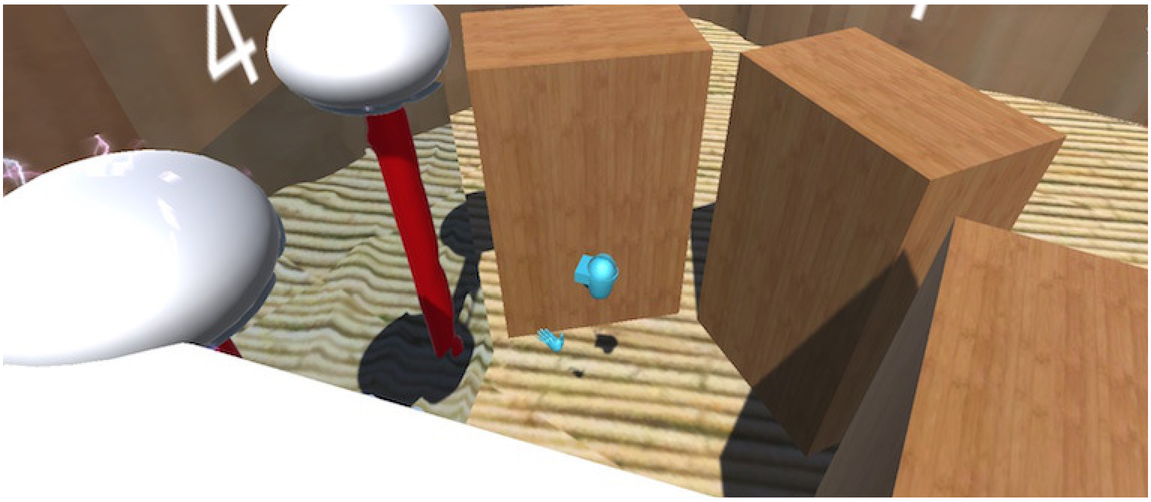

Figure 36: Study participant in the virtual world of the study, surrounded by five virtual walls; each wall embodies a study condition.

3.3.3.2 Apparatus

The apparatus was the prototype described earlier and shown in Figure 41. Participants wore EMS and vibro-tactile actuators, one or both of which would be activated according to interface condition. To reduce setup time, we actuated only *one* of the participants' hands. Also, since the set-up offered only vertical surfaces, we further simplified the set-up and used only electrodes on the biceps and wrist extensor muscle.

3.3.3.3 Task and procedure

Participants were prepared and placed into the virtual world shown in Figure 36. For each of five trials, the experimenter instructed the participant on which of the five walls to explore. Walls were labeled with numbers for that purpose. After touching the respective wall design for about 30 seconds, allowing for between 5 to over 20 touches, participants verbally rated this wall, the experimenter writing it down. The order of the five interface conditions was randomized for all users prior to the start of the study.

3.3.3.4 Participants

We recruited 13 participants (4 female, 22.4 ±2.1 years). Six participants had previous experience with VR headsets and 5 had previously experienced EMS.

3.3.3.5 Hypotheses

Our hypotheses revolved primarily around the *repulsion* wall, the *soft* wall, and the *vibro only* baseline. (**H1**) The *repulsion* condition would be perceived as more realistic than the *vibro only* condition. (**H2**) The *soft* condition would be perceived as more realistic than the *vibro only* condition. (**H3**) The *soft* condition would be rated more realistic than the *soft wood* condition. (**H4**) The repulsion condition would be considered more impermeable than the *vibro only* condition.

3.3.3.6 Results

This section quotes Bonferroni-adjusted *p*-values.

Preference. Eleven participants stated a preference for one of the EMS-based interfaces; only to two participants preferred *vibro only*. As depicted in Figure 37, eight participants picked the *repulsion* wall as their favorite. Another three listed this interface as their second favorite.

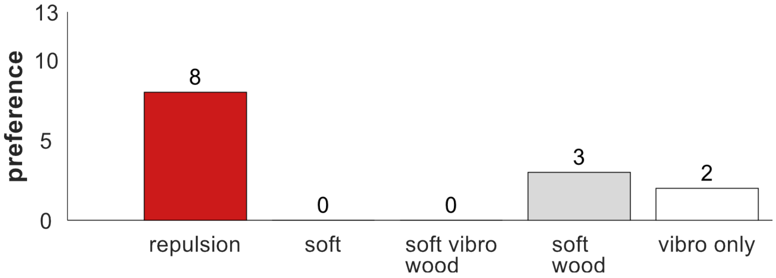

Figure 37: Eight participants picked the *repulsion* wall as their favorite design.

This suggests that our *repulsion* wall was particularly well designed. This raises the question on what aspect of the repulsion design caused this preference. One explanation might be found in participants' assessment of the realism of this design.

Realism. Figure 38 shows how participants rated the five conditions with respect to the question "what I feel matches what I see." A repeated measures ANOVA (as suggested by [113]) found differences between conditions ($F(4,48) = 6.22$, $p = .000$).

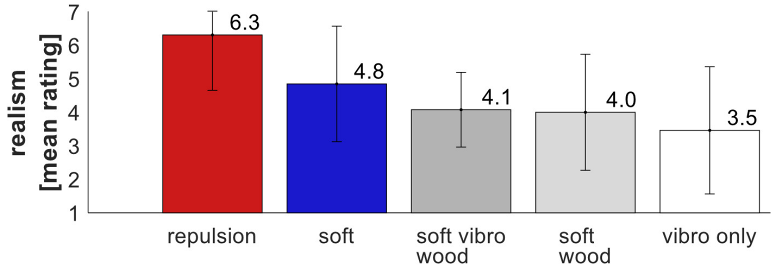

Figure 38: Participants rated the *repulsion* wall as the most realistic.

As expected, the repulsion condition received higher ratings than all other conditions, significantly so for *soft vibro wood* ($t(12) = -5.06$, $p = .002$) and *vibro only* ($t(12) = 3.71$, $p = .030$) and with a strong trend with regard to *soft wood* ($t(12) = -3.38$, $p = .055$). This confirms our hypothesis **H1.**

While the *soft wood* condition was rated higher than the *vibro only* condition, this difference was not found to be statistically significant. Hence, **H2** was not supported**.** Even though there certainly was a trend, the differences between the *soft* condition and the three conditions that visually display a wooden texture were not found to be statistically significant. We therefore found no support for hypotheses **H2** and **H3**.

The second possible explanation for participants' preference for the *repulsion* wall might be found in that design's performance.

Impermeability. Figure 39 shows participants' assessment of "this wall was able to prevent me from passing through". A repeated measures ANOVA found significant differences between conditions (*F*(4,48) = 6.68, *p* = .000). The main finding here is that *repulsion* was rated as more impermeable than *vibro only* (*t*(12) = 4.18, *p* = .013), which confirms our hypothesis **H4**.

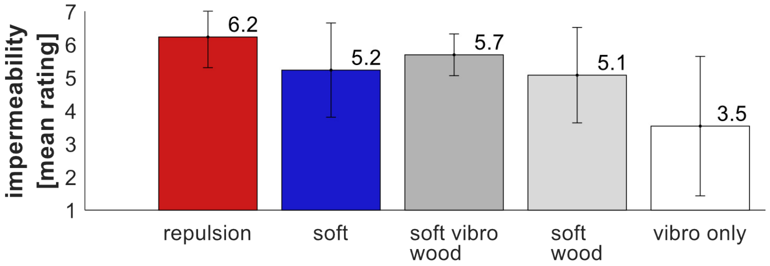

Figure 39: Participants rated the EMS conditions more impermeable than *vibro*.

These responses are backed by our measurements on how deeply participants penetrated into each wall (Figure 40, measured using our optical tracking system *Optitrack 17w*).

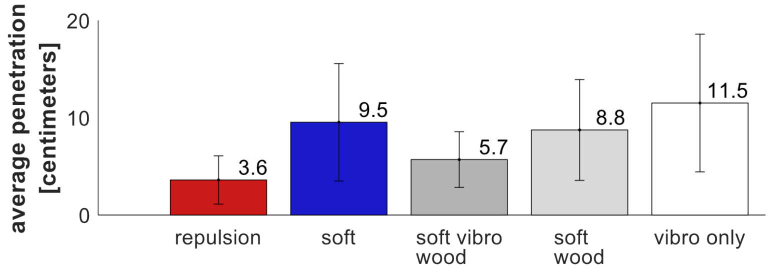

Figure 40: The *repulsion* wall stopped participants' hands on average 3.6 cm (error bars denote std. dev.).

A repeated measures ANOVA found a significant difference between conditions (F(4, 48) = 7.72, *p* = .000). The *repulsion* wall stopped participants' hands at only 3.6 cm on average, which is significantly earlier than *vibro only,* as a post-hoc t-test revealed (*t*(12) = -3.54, *p* = .040). This provides additional support for our hypothesis **H4**.

Participants penetrate the *soft* wall significantly deeper than the *repulsion* wall (*t*(12) = 3.69, *p* = .031). This, however, is expected, given that it had been designed to allow for 10 cm penetration. Data on the average penetration in the virtual barriers was normal according to Shapiro-Wilk tests. The assumption of sphericity was not violated ($\chi 2(9)$ = 14.42, *p* = .11).

3.3.3.7 Participants commentaries and open-ended questions

All participants stated that the EMS was fitting the expectation of the "electro wall" visuals and noted its effectiveness in stopping them. P8 added "the pushing effect from these (virtual walls) felt like how a real wall pushes back". Another participant, P1, added, "EMS matched the springiness I expected of the wobbly wall". P5 remarked "it is funny because I feel (the wall's force) it but I know nothing is there".

One participant (P2) emphasized that the “EMS tingles and hence reveals the source of the force”, yet, P7 stated “I did not realize how my hand was moving, I could not tell it was the EMS”. Furthermore, three participants emphasized that the vibration did not match the expectations of the wooden walls.

3.3.3.8 Summary and discussion of findings

Our first study confirmed several key hypotheses. First, it found that *any* of the EMS-based designs performs better than the most commonly chosen haptics option today: vibrotactile feedback.

The *repulsion* design did particularly well. It was rated the most *impermeable*, suggesting that it is suitable wherever virtual world designers have to stop users from passing through. The *repulsion* design also scored highest in terms of consistency between visuals and haptics. This matches our observation that the optical and acoustic effects behind this design work particularly well at covering up the EMS actuation, especially given that it is brief. Finally, and arguably most important, the majority of participants picked repulsion as their favorite design.

The *soft* design, while not as strong as the repulsive design, demonstrated good “all-round” qualities. Reasonably realistic and reasonably impermeable, it outperformed vibrotactile. The combination with vibrotactile does not seem to lower performance, but does not add much, either.

3.3.4 Implementation details

To help readers replicate our design, we now provide the necessary technical details.

3.3.4.1 EMS hardware and calibration

We used the battery-powered, medically compliant 8-channel muscle stimulator (*Hasomed Rehastim* [21]) depicted in Figure 41.

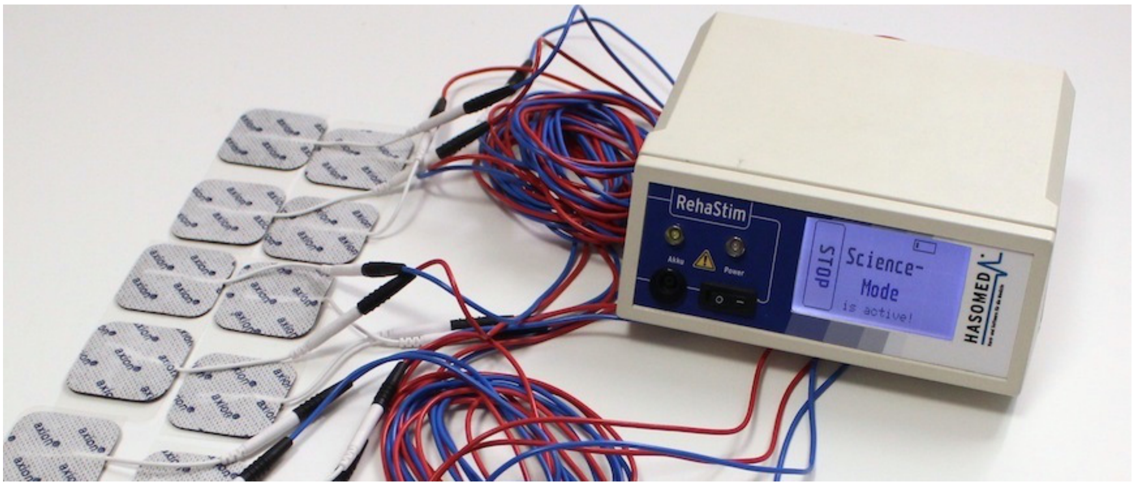

Figure 41: The battery powered, medically compliant, muscle stimulator we used.

The stimulator is designed for EMS and outputs up to 100 mA per channel. The stimulation is triggered via serial commands using Hasomed's custom protocol, which we generate from inside Unity3D. For each EMS channel, we keep the intensity constant and instead modulate the pulse width (in µs), which we henceforth denote as PWM. This allows us to have greater control (sub-mA) than by varying the current.

Like any haptics system based on EMS, our system requires calibration prior to use. In order to determine what is comfortable for a particular user, we continuously increase the current until we observe a small movement of the targeted muscle. We then let the user calibrate an upper bound that is still comfortable and pain-free. We perform this procedure in the order palm/wrist extensor, biceps, triceps and shoulder rotator muscles.

3.3.4.2 Electrode placement

Figure 42 depicts the electrode placement we used to actuate the user's arm and hand. We placed electrodes on the following muscles: (a) palm and wrist extensors (covering both the *extensor digitorium* and *extensor carpis ulnaris*), (b) biceps, (c) triceps and (d) shoulder external rotators (covering both the *infraspinatus* and *teres major/minor*).

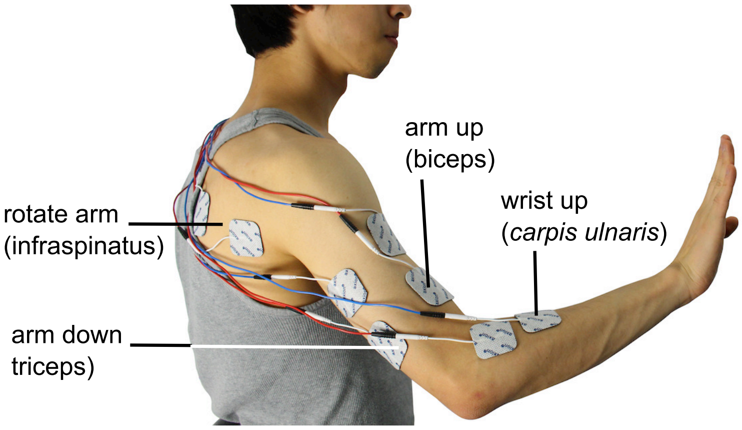

Figure 42: Electrode placement for arm and shoulder.

3.3.4.3 EMS parameters

Each individual haptic effect, such as the *repulsion* of a wall or the sensation of *picking up* a box has its own EMS settings. Values vary across users and should be customized using the calibration procedure described above. Still, we provide a data set example, gathered from one participant in our experiment:

1. **Repulsion wall**: palm extensor at 17 mA, 100 µs PWM for 300 ms, biceps at 15 mA and 200 µs PWM for 300 ms.
2. **Soft wall**: palm extensor starts at 15 mA and 75 µs PWM, biceps at 15 mA and 70 µs PWM. Here, the stimulation increases linearly as the user presses into the wall or button (function of the distance to the center of object, e.g., a button). The maximum values are 100 µs PWM for the palm extensor and 175 µs PWM for the biceps.

3. **Picking up a box**: shoulder muscles at 20 mA and 250 µs PWM and triceps at 15 mA and 150 µs PWM.
4. **Pushing a box backwards**: palm extensor at 15 mA and 100 µs PWM, biceps at 15 mA and 150 µs PWM.
5. **Pushing a box sideways**: palm extensor at 15 mA and 200 µs PWM and biceps at 15 mA and 110 µs PWM.

All the aforementioned effects except *repulsion* use a simple linear mapping between the EMS intensity and normalized distance to the object [0=center, 1=surface]. Our intensity-distance mapping is defined as:

$$(\min(\mathrm{intensity}_{\mathrm{maximum}} * (1 - \mathrm{distance}_{\mathrm{normalized}}) * \mathrm{growth}_{\mathrm{factor}}) \\ + \mathrm{intensity}_{\mathrm{offset}}, \quad \mathrm{intensity}_{\mathrm{maximum}})$$

When *pushing a box backwards* or *sideways*, the applied mapping is steep (hence, growth factor is 2) since the effect should be strong upon contact with the object. However, in the *Soft Wall* the intensity build up is softer, hence the growth factor is kept at 1. When *picking up a box* the growth factor used is also 1; note that on the triceps, the intensity offset is calibrated higher than for the other effects, as to constantly simulate the cube's weight as soon as the user grasps it. Lastly, the *Repulsion* effect does not utilize such a mapping because it is only active for 300 ms.

3.3.4.4 VR engine

We implemented our virtual worlds in *Unity 3D*.

3.3.4.5 Tracking

We track the user's headset and hands using rigid body optical markers and cameras (8x *Optitrack's Prime 17W*) covering a tracking volume of 4.5 x 4.5 x 3m. When replicating our system, an HTC *Vive* would be equally suitable.

Our system determines collisions between the user's hands and virtual objects using collider objects in Unity. This triggers the respective muscle stimulation patterns by sending a message to a server application that communicates to the EMS device.

3.3.5 Example widgets

The *soft* design and the *repulsion* design together allow us to create the haptics for a reasonably wide gamut of virtual objects. In order to illustrate this, we have created a set of example objects and widgets. We combined these widgets into a simple virtual world, which forms the basis for our second user study (see below).

We use the *soft* design for the majority of objects. This design allows users to make physical contact with the object, as the user's hands partially penetrate into the object's interior. This helps maintain physical contact with an object, making it possible to drag or carry objects around.

We use the *repulsion* design to complement the soft design and, in particular, we use it to implement those walls, doors, and windows that are designed to prevent users from passing through, such as in a jail-like surrounding.

We continued to use electro visuals for the repulsion design. For the soft design we used a multi-layered translucent texture. In order to allow us to apply these designs to arbitrary objects, we implemented their visuals in the form of translucent textures.

We now demonstrate these widgets by giving a walkthrough of the simple virtual experience we designed for the second study. This VR experience consists of three rooms connected by three hallways. Inside this world, everything users can reach is complemented with a haptic effect either based on one of the two designs, or based on an interpolation between the two.

3.3.5.1 Room 1: Repulsion.

We designed the first room so as to primarily illustrate the repulsion design.

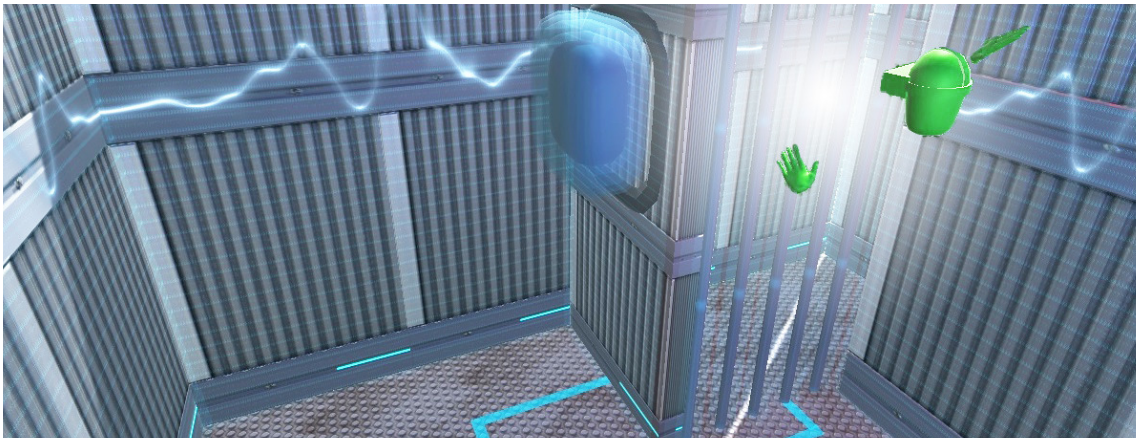

Figure 43: The jail cell features "electrified" walls and a gate. Touching any of these repel the user's hand.

Repulsion walls: As illustrated by Figure 43, this room is designed as a jail cell with an "electrified" gate and walls. When touched, these repel the user's hand, which is accompanied by the sound and visuals discussed earlier.

Button: Figure 44: A button allows users to raise the gate. The button is soft, allowing the user's hand to penetrate its surface. The system accompanies this with a sense of increasing counter pressure. The button then tracks with the user's hand and the counter force stays constant until the button is all the way "in", at which point the counter force increases substantially.

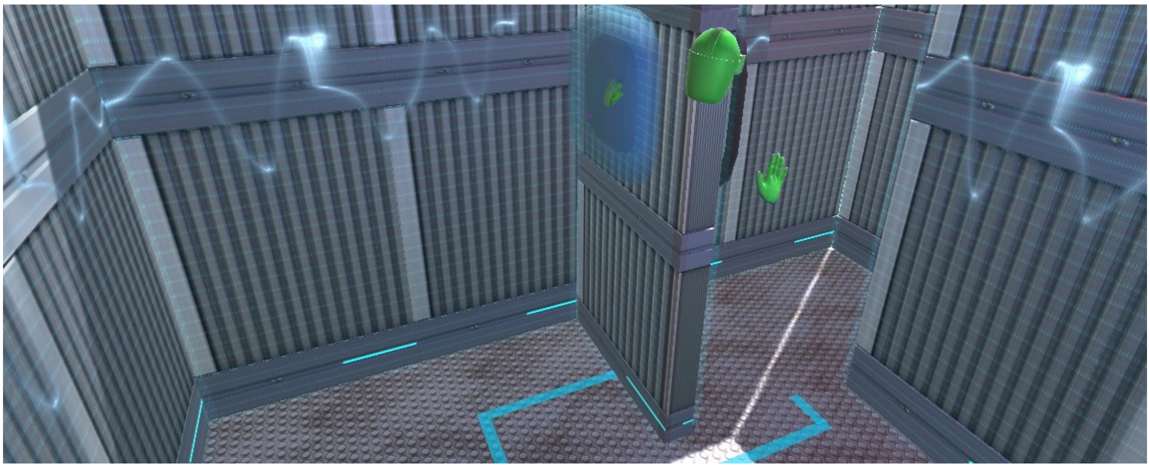

Figure 44: Pushing this soft button opens the gate.

Projectiles: As users rush down the hallway, a security system shoots projectiles at them, which they fend up with their hands. The system renders the 12” projectiles using a strong repulsion effect.

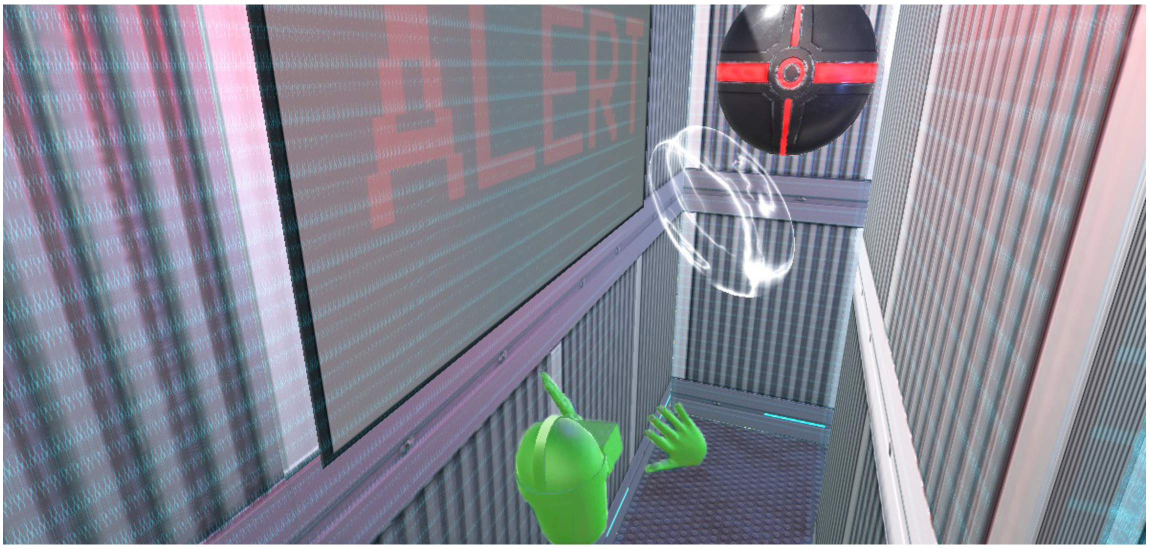

Figure 45: This cannon shoots projectiles that we render using the *repulsion* effect.

3.3.5.2 Room 2: Selected widgets

We use this room to illustrate some traditional GUI widgets rendered as interface elements in VR, in particular a slider and an analog rocker switch.

Slider mechanism: As shown in Figure 46, users operate a pair of sliders in order to align the pipeline elements that establish a hydraulic link. The sliders’ knobs are based on the soft design. The knobs protrude far enough to allow operation from multiple angles. If users operate the sliders while *facing* the wall, the system primarily actuates their shoulder muscles; if they turn *parallel* to the wall, the system interpolates from shoulder muscle to biceps.

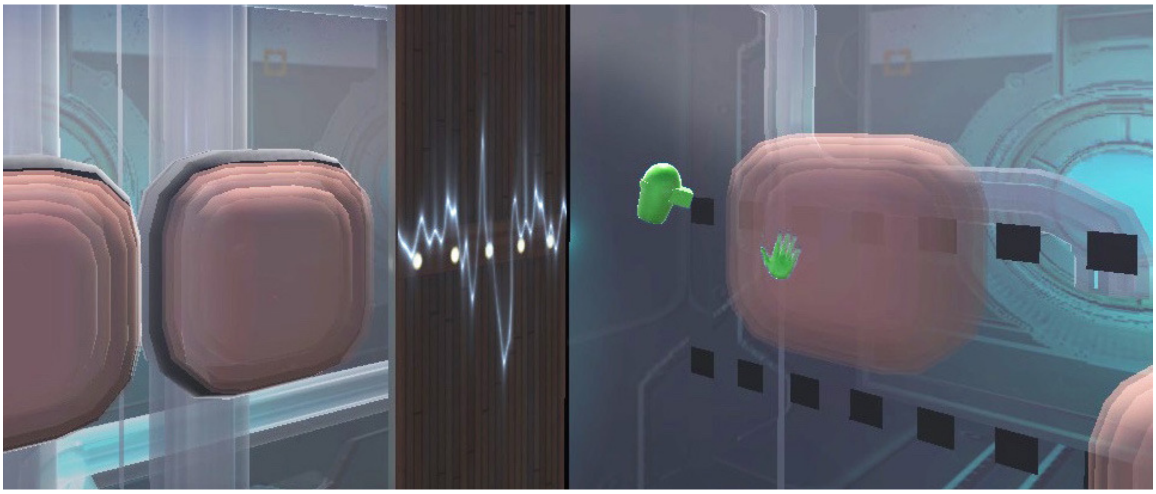

Figure 46: The user is dragging the knob of a slider mechanism.

Widgets that push back: Users can operate the pump to re-establish pressure in the hydraulic system. The device consists of two buttons connected by a rocker mechanism (left side of Figure 46), i.e., as users push one of the buttons in, the button comes out and pushes against their other hand, providing a simple animated haptic response.

3.3.5.3 Room 3: Lifting, punching, and throwing

We use this room to illustrate moveable objects.

Pushing, lifting, and dropping objects: As shown in Figure 47, there are two cubes. As users push the first one towards the adjacent button, they feel haptic feedback. Depending on the angle of attack, the feedback actuates users' biceps, shoulder, or both.

Figure 47: Two cubes that users can push onto the button on the right or pick up.

If users *pick up* a cube with both hands (Figure 48), they feel a resistance when their hands come together as to grasp the cube. This happens because their shoulder muscles are stimulated as to open their arms outwards.

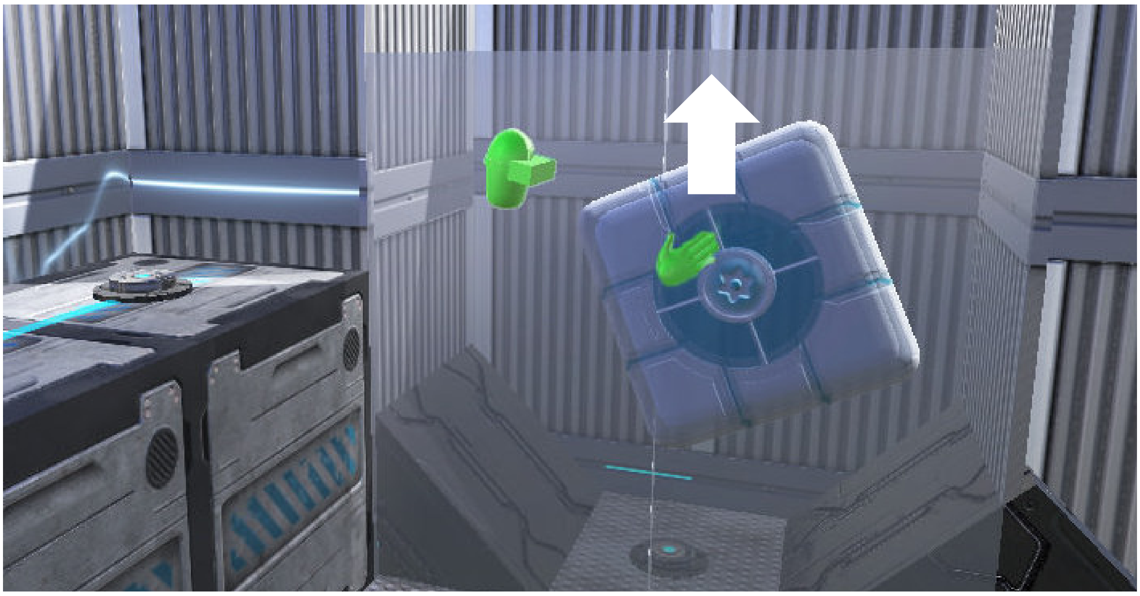

Figure 48: The user picked up a cube and is about to throw it over the glass.

Furthermore, users also feel the weight of the cube in their triceps, as discussed in the introduction. In Figure 48, we see how users pick up the cube and throw it over a glass wall, down a chute, which activates a second button.

Punching objects: Figure 49 shows a third cube that rests on a slide, which leads up to the last of the three buttons. If users push this cube they feel a soft effect in response. However, the contraption requires users to *punch* the cube up the slide in one go. The harder they hit the membrane, the more the system shifts its haptic response from *soft* effect to *repulsion* effect.

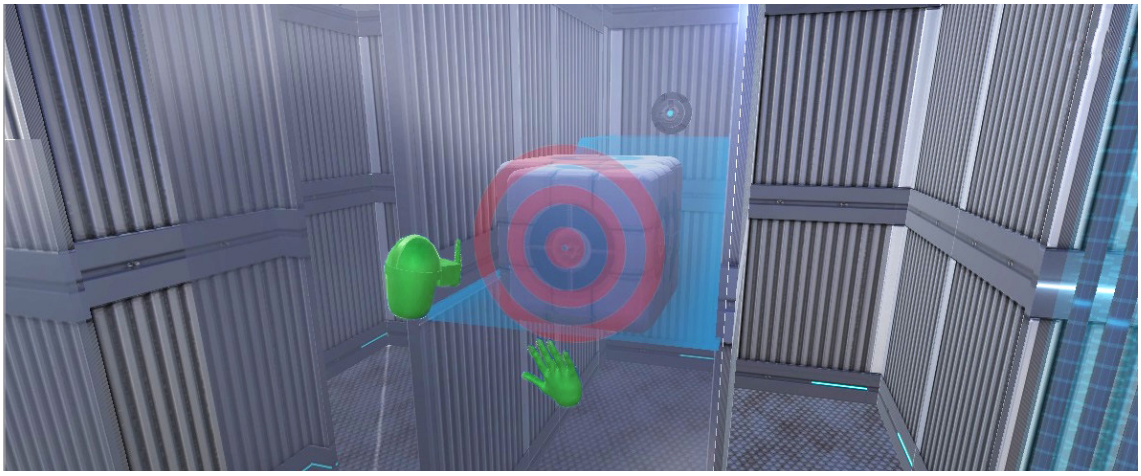

Figure 49: Users have to drive this cube up the ramp by punching it. The system responds with a mix between a soft effect and a repulsion effect.

3.3.6 User study 2: the experience

Given that our first study was very focused on comparing different wall designs, we now wanted to see what an actual virtual reality experience combined with our EMS prototype would be like. We thus conducted a second study. This time, we gathered only a minimum of Likert scale data, as we were mostly interested in participants' open-ended feedback.

3.3.6.1 Apparatus

Participants wore the same general type of EMS apparatus as in the first study. However, this time we actuated not only biceps and wrist, but also shoulders. We also considered both hands, for a total of 6 actuation points (compared to 2 in the first study). In exchange, we left out the vibrotactile actuator, which our first study had found to be of only limited usefulness.

3.3.6.2 Task and procedure

Participants were outfitted with our EMS device and then underwent the same type of interactive calibration procedure as in our first study. We then placed participants into Rooms 1 and 3 of the virtual world previously described.

There were two interface conditions. In the *EMS condition* the EMS equipment was *on*. In the *baseline* condition the EMS equipment was *off*. Participants thus went through the experience twice (in counterbalanced order). After each run, they filled in a questionnaire.

3.3.6.3 Participants

We recruited 6 new participants from our institution (1 female, 22.0 ±2.09 years old). Five participants had experienced VR headsets before and two had experienced EMS before.

3.3.6.4 Hypothesis

We hypothesized that our EMS prototype would lead to a better user experience than the control condition without.

3.3.6.5 Results

As illustrated by Figure 50, the *EMS* condition received substantially higher ratings. As a matter of fact, *all* participants rated *EMS* higher than *baseline*. This confirms our main hypothesis.

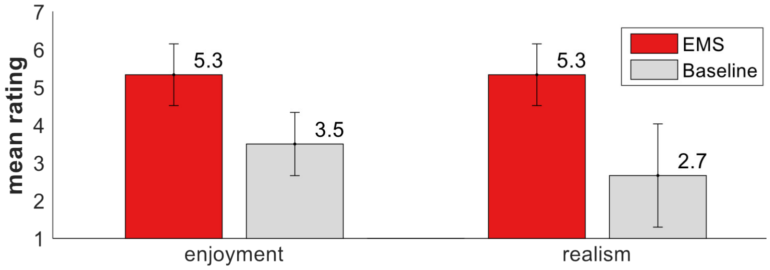

Figure 50: Participants rated the *EMS* condition higher than in baseline.

Participants also responded to “which object, widget, or effect in the virtual world did you like best and why?” (Figure 51). In the EMS condition, 3 participants preferred the walls (“electrified walls”), 2 participants preferred the cube and one the button. For the control condition, 4 participants preferred the cubes and 2 the button; in the control condition no participant preferred the walls.

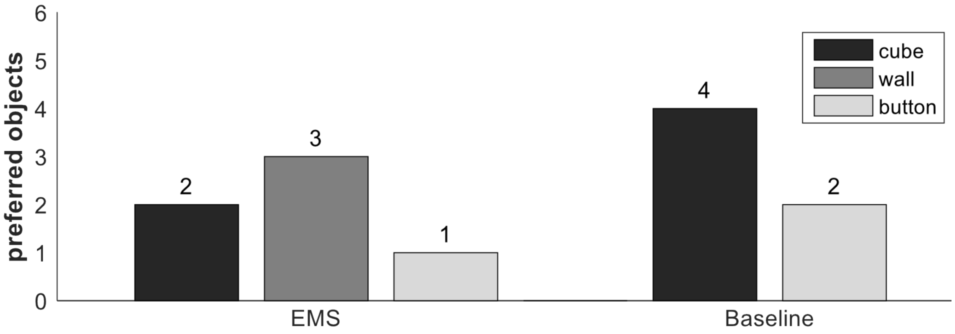

Figure 51: Participants preferred the experience in the *EMS* condition.

We invited participants to comment on their experience. P1 said, regarding the electro wall, “I can’t go through this wall”, when pointing out the soft walls P2 remarked, “it feels less real (than electro wall) because like this (softer EMS) I can go through”. P1 concluded, “I much prefer the sensation *with* EMS, otherwise there is nothing there, just air”.

P2 added that the electro walls “added more experience, like better gameplay” and remarked that the “the soft walls were not always necessary, maybe not all switches need it”. P2 also added “the cannon (was great) too, but it felt more like a punishment from passing (in front of it)”.

P3 stated, “I don’t know how VR is supposed to feel but without the EMS it did not seem real at all”. Then added, “Also… I could not really grab objects without the EMS, because nothing was there, my hands touched each other”. When choosing the button as their favorite object for the baseline condition, P3 remarked, “nothing felt real (without EMS) but at least the button was the only one that was at least kind of realistic”. P3 also added that “I immediately felt the difference between an electro wall and a soft wall”.

P4 added “The electrified walls worked great, it really felt like touching one. Really surprising.” P4 also added “I liked punching the cube at first (without EMS) because it is very (physically) involving, then when I tried it later (with EMS), I felt some impact force; it was great.” Later P4 added “the cube is the hardest one to believe, because there are many ways to hold it and it and (the EMS) does not always work that well.”

P5 commented with respect to the EMS condition “(The wall) worked great, this was the most realistic element I tried”. When P5 experience the same wall without EMS, P5 stated, “oh my… these walls feel really boring”. P5’s expressed that EMS contributed to simulating walls, buttons, projectile hits, lifting and throwing cubes, but not to punching. P5 explained “I would also like to feel something in my hand, not just the muscles of the arm, it feels misplaced.”

P6 stated, “(the) button works great and so do the walls. It feels just right, if I push it, it pushes back and the feeling is continuous”. P6 commented about the baseline condition “only the cubes feel right to me because I can manipulate them, the walls and buttons feel wrong”. P6 continued “EMS really helped me feel the walls, the button and the projectiles, those really felt strange without EMS, like energy that went nowhere”. Lastly, P6 added “the stimulation while operating the cube worked well, but I would have preferred it to push downwards, like pretending to have weight.”

3.3.6.6 Summary and discussion of the study 2 findings

As expected, EMS added to participants’ experience. In particular, participants’ responses showed that the “repulsive” wall design using EMS did a good job simulating walls in VR.

We also observed, how these walls really stopped participants and participants described them as realistic. In contrast, all participants seemed to agree that walls were *not* realistic in the no-EMS baseline condition. Furthermore, based one participant’s feedback regarding the cube (“I would have preferred it to push downwards, like pretending to have weight”) we added stimulation to the triceps muscles, hence rendering the cube’s weight.

3.3.7 Conclusions on EMS for VR walls & heavy objects

On a more general note, this investigation confirms what we also found in impacto: the key design principle to attain realism while using EMS is the *justification* of the force; in other words, it is the designer’s job to mask out that the force feedback is coming from the user’s own muscles. While in impacto we did this using a solenoid that delivered a realistic tactile sensation, in VR walls we explored adequate visual metaphors to achieving this, for instance: the “electrified wall”, the “magnet” or the “soft cubes”.

Our research so far, has informed us that EMS, when designed carefully, is a suitable alternative to traditional motor-based haptics. In particular, by building these three prototypes we found that EMS setups can be lightweight and small enough to be mobile (as in muscle-propelled force feedback) or wearable (as in impacto). However, because our main claim has been that EMS is wearable and harmonizes well with, contemporary, real-walking VR, we missed exploring a crucial opportunity that EMS lays out before us, which was hidden in our first project: *EMS leaves the user's hands free to interact with the real world*, i.e., coincidentally, in muscle-propelled force feedback the user was comfortably *holding* a phone.

3.4 Hands-free force feedback for mixed reality

We now explore this newly found benefit of EMS-based force feedback: keeping the user's hand free to explore the real world. This benefit is particularly useful in the contexts of augmented reality (AR) and mixed reality (MR); in which, unlike in VR, the user's interactions take place in a world surrounded by real objects, functional appliances, and so forth. Thus the main challenge for MR haptics is that users may encounter not only virtual objects, *but also physical objects*. This means that haptic technology for MR must leave the users' hands unencumbered [158]. Furthermore, MR users may want to avoid any kind of bulky motor-based devices, as these tend to be visible in MR and might even occlude the real-world objects users are interacting with.

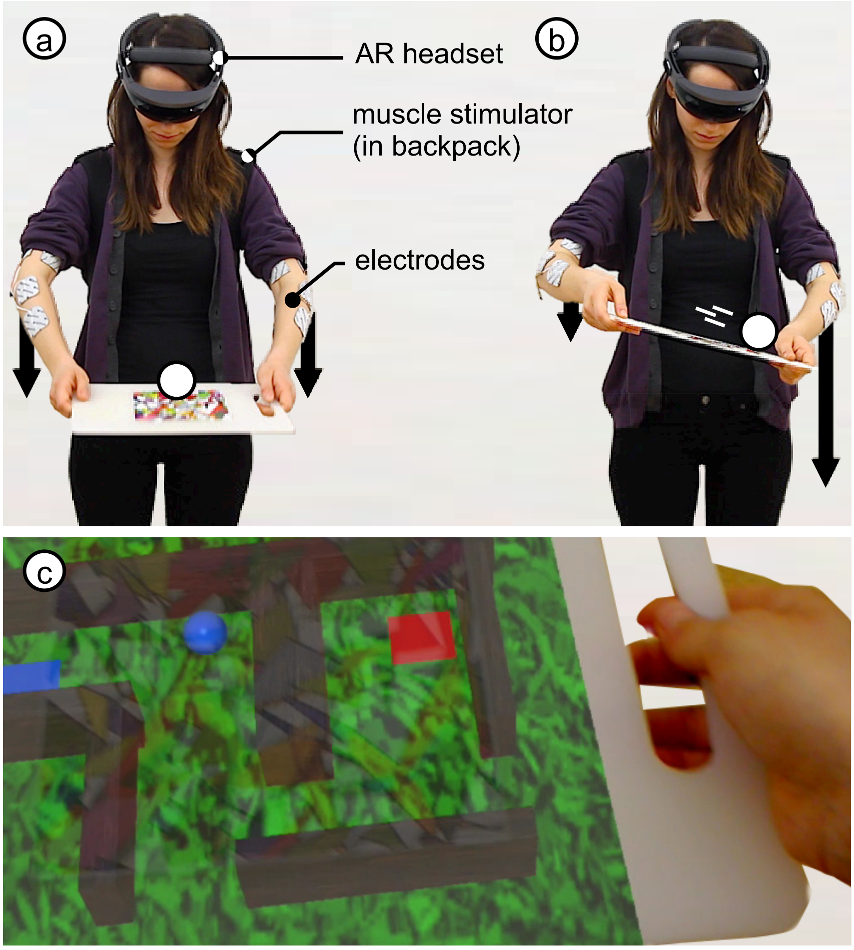

Figure 52: In this mixed reality game that uses a physical tray as prop, our mobile EMS system renders shifts in the tray's center of gravity as the marble moves.

Instead, in this investigation, we add force feedback to MR games and experiences using electrical muscle stimulation; an approach that is small, light, and fits under the user's clothing. Figure 52 illustrates this at the example of our balance marble game using a physical tray as a game prop, which our approach augments via EMS-based force feedback. Our MR system leverages EMS to allow this game to render haptic effects, such as the shift in the tray's center of gravity as the virtual ball moves on it.

We demonstrate, by means of four simple mixed reality experiences built for the Microsoft HoloLens, how EMS supports force feedback not just in VR (as our previous research demonstrated) but also in the broader spectrum of Reality. This includes adding force feedback on a variety of situations rooted in the Reality-Virtuality continuum [105], ranging from interacting with purely virtual objects, to passive props and augmented physical devices.

3.4.1 Walkthrough of our mixed reality experience

Figure 53 shows a user wearing our EMS for mixed reality system. The user is wearing a Microsoft HoloLens headset that runs our stand-alone and untethered MR experiences, for instance, this interior design application. The user is exploring what light bulb might illuminate her painting best. She uses a regular cup as an impromptu tangible brightness dial to explore different light settings. The HoloLens displays the associated GUI. As she tries to increase brightness past the allowed maximum, the system actuates her wrist rotation muscles by means of EMS, preventing her wrist to go past the maximum and she hits a hard stop.

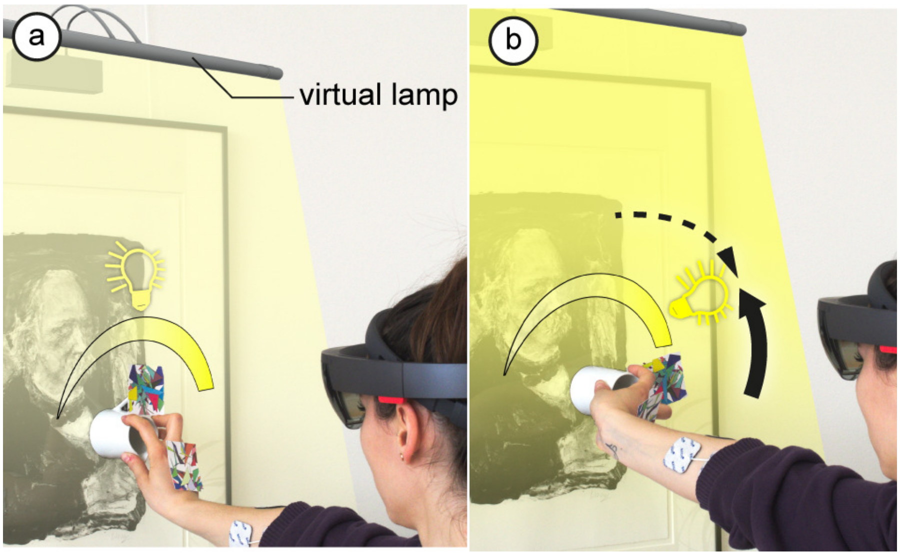

Figure 53: Using a cup, actuated by means of EMS, as a tangible brightness dial.

Figure 52 also depicts the system she is wearing, i.e., our EMS signal generator connected to electrodes placed on the user's wrist, arm, and shoulders. Our system leaves her hands free to interact with physical objects at all times.

Figure 54 depicts the user's view through the HoloLens. In the interest of visual clarity, we show subsequent images in a 3rd person perspective. Figure 54b highlights one of the markers we are using in *some* of our examples; this was a stopgap measure to obtain the *orientation* of props and of the user's hand (see "Implementation details").

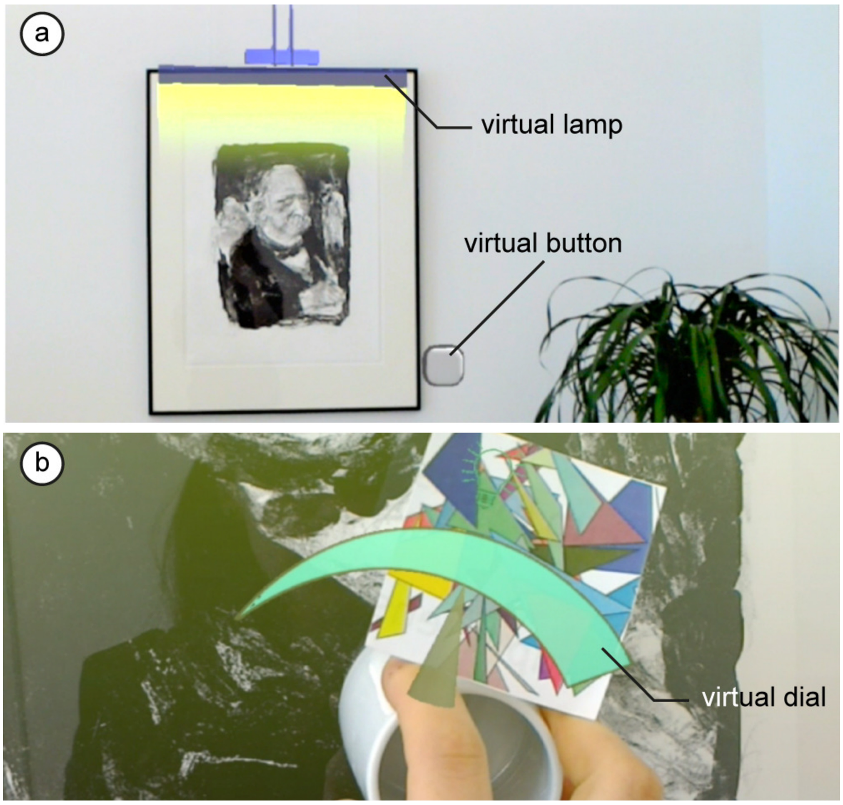

Figure 54: The previous scene through the HoloLens.

. The next figures show a slightly longer walkthrough of the interior design experience. Here, our user is configuring furniture for her future living room, which includes a couch and a lamp specifically tailored to highlight a valuable painting on the wall. We designed the walkthrough to showcase examples from three classes of objects on the reality-virtuality continuum [105], ranging from fully virtual to physical objects.

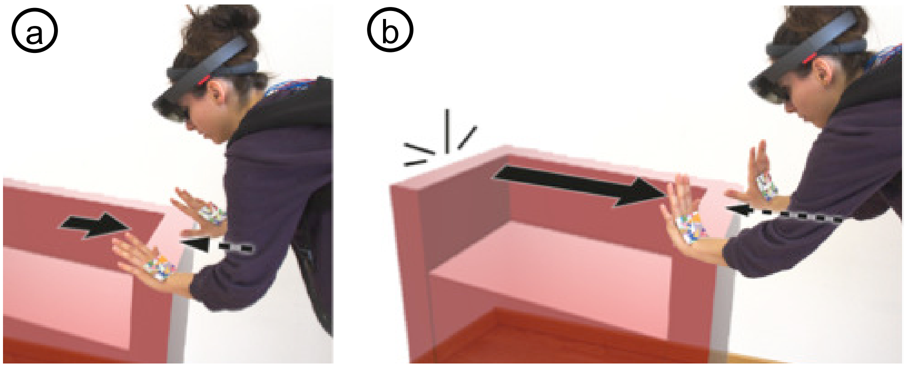

Figure 55: (a) This user physically drags the couch and feels the EMS simulated friction against the floor. (b) As the couch collides with a real wall, the system stops the user by pushing the user's shoulders and wrists backwards.

1. Providing virtual objects with force feedback. Figure 55 shows the user exploring different placements in the room by pushing a couch with her two hands. Our system renders the couch's friction against the ground as a gentle force pushing her hands back, by stimulating the user's shoulder muscles (Figure 55a). Lastly, when the virtual couch hits the real wall, the EMS force feedback increases and informs the user of the collision (Figure 55b).

Virtual mechanisms. As illustrated by Figure 56, the user now explores a lamp from the catalog. After placing gallery lighting so as to highlight the painting on the wall, the user turns the lamp on by pressing the switch. Here our system renders the button's mechanics using force feedback.

Figure 56: Our EMS system renders the forces of the button's mechanism.

When the user presses the switch to turn the lamp on, she feels the constant counterforce of the spring inside the button until the button is fully depressed and it latches (Figure 56b). To achieve this effect our device primarily actuates the user's wrist, complemented as well with some light actuation on the shoulder.

Also, when pushing the switch while it is on the "on" position (button depressed inside), our system applies a weak force feedback until the switch is pressed. The virtual spring then releases at full force as the user's hand is pushed backwards. This is an example of how EMS recreates the expected physics of objects (e.g., a spring and latch), so as to better align the virtual and the real in mixed reality.

2. Turning passive props into impromptu tangibles using EMS. The user now configures the brightness of the lamp. As shown in Figure 57, the user picks up a cup to serve as a tangible brightness dial. The system extracts the dial's rotation and maps it to the brightness of the lamp. The lamp's intensity is displayed as a halo visually projected onto the scene. Our system adds physical constraints, preventing the user from rotating past the minimum and maximum values. We render these constraints by stimulating the user's wrist rotator muscles so as to provide a counterforce to the user's direction of turning, thereby stopping the user's twist.

Figure 57b shows how our system also adds detents to the dial. These detents inform the user that the selected intensity is available as a light bulb, while values in between require an extra dimmer. Our system renders the effect by stimulating the antagonist muscle with brief pulses, i.e., when the user rotates the dial clockwise the system sends short pulses that turn counter-clockwise and vice-versa.

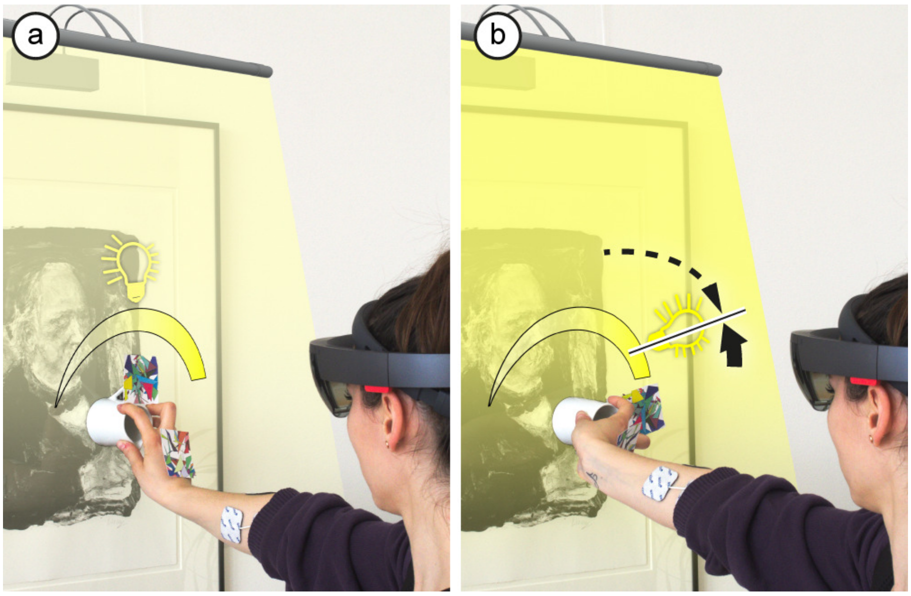

Figure 57: The user configures the intensity of the desired light bulb using a cup as a stand-in for a dial (passive prop). Using EMS force feedback, our system augments the tangible with constraints and detents.

Physically linked passive props. Now, the user realizes this light's color temperature is too warm for her artwork. Hence, this time, she places two cups, the left represents light temperature and the right one stands for intensity (Figure 58a).

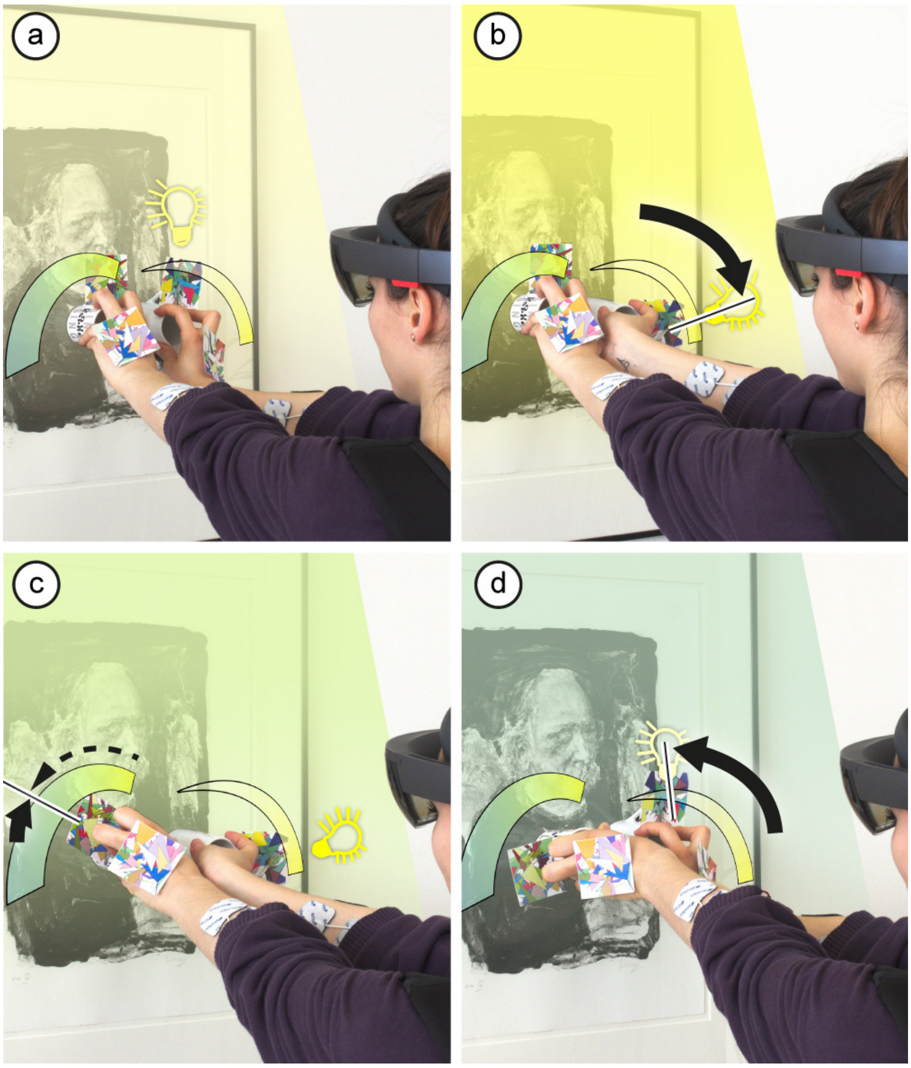

Figure 58: The user manipulates two cups to control the light temperature and intensity simultaneously.

Since she previously left the intensity value at maximum, as soon as the system recognizes the cup, it actuates her wrist rotator muscles to place the dial back in the last used value (Figure 58b). We do so to align virtual and physical worlds into a coherent mixed reality. Then, she explores combinations of different light color temperature and intensities bimanually. These two parameters are dependent in that bulbs are available only in certain intensity-temperature combinations. Changing one parameter (e.g., light temperature) causes the system to adjust the other (e.g., intensity) to achieve an existing option in the catalog. In the Figure 58c, for example, the user chooses a “colder” light, causing the system to switch to a less intense bulb by actuating the user’s wrist as to reach that option (Figure 58d).

3. Augmenting real objects. Finally, the user decides whether to upgrade her thermostat to a better model offered in the catalog. The better thermostat offers detents that inform users about the most recently used temperature setting. Here, our system simulates the new thermostat by virtually enhancing the existing thermostat with the missing feature. The user explores this new "version" of the thermostat and decides to order one.

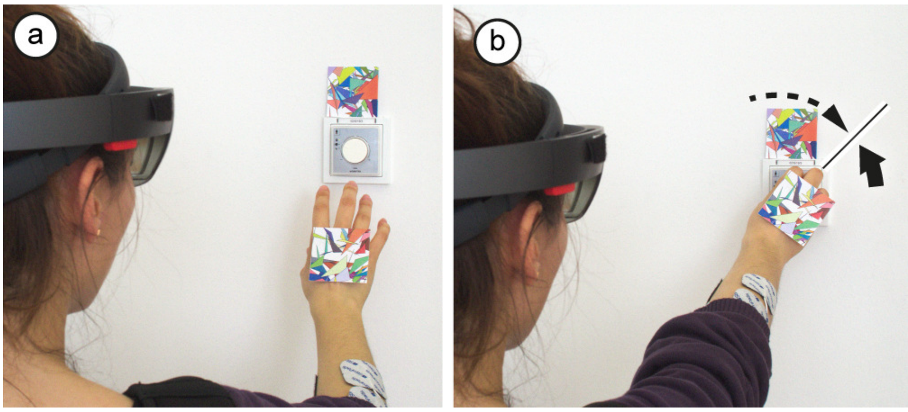

Figure 59: Here, our system enhances a fully functional thermostat with detents.

3.4.2 Summary of walkthrough

As mentioned earlier, the examples in our walkthrough were chosen to illustrate how our system covers Milgram et al.'s reality-virtuality continuum [105] (see Figure 60).

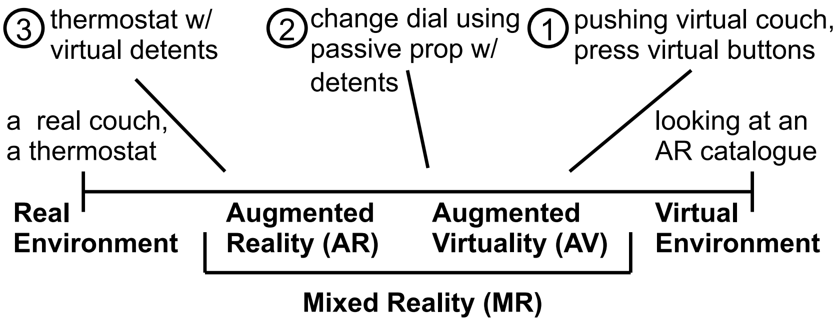

Figure 60: Walkthrough examples mapped to the reality-virtuality continuum.

Starting from the right, we (1) provided EMS-based force feedback to fully virtual objects, here a couch. (2) We then provide EMS force feedback to objects augmented with passive props. We used this to provide these objects with additional physical properties, here detents and constraints to a dial. (3) Lastly, we augmented a physical device with an additional property, here a thermostat enhanced with detents.

3.4.3 Demonstrating EMS haptics in mixed reality experiences

As discussed earlier, we think of this as a system that enables physical feedback in mixed reality, not only limited to the aforementioned walkthrough scenario (i.e., the interior design application). To emphasize this point, we designed an additional three immersive MR experiences in which we made use our system for the missing force feedback; each of these applications highlights a particular step of the reality-virtuality continuum:

1. Purely virtual: shooting the catapult. Figure 61 shows a simple MR game featuring a virtual catapult that appears in the user's physical surroundings. (a) The user arms the catapult by pulling the catapult's bucket backwards. We implemented this example without markers, instead using HoloLens' "pointing". As the user pulls the bucket backwards, our system provides force feedback that simulates the increasing tension of the catapult; this force feedback also serves as an indicator for the user on how far the projectile can be expected to fly. Our system achieves this by stimulating the antagonist muscle (triceps) proportionally to the catapult's arm angle. (b) When the user opens the hand (which terminates the HoloLens gesture) the catapult releases. Our system abruptly ceases to stimulate the triceps ceases and the remainder force in the user's biceps, which now does not have a counter force, creating a sensation of actual recoil.

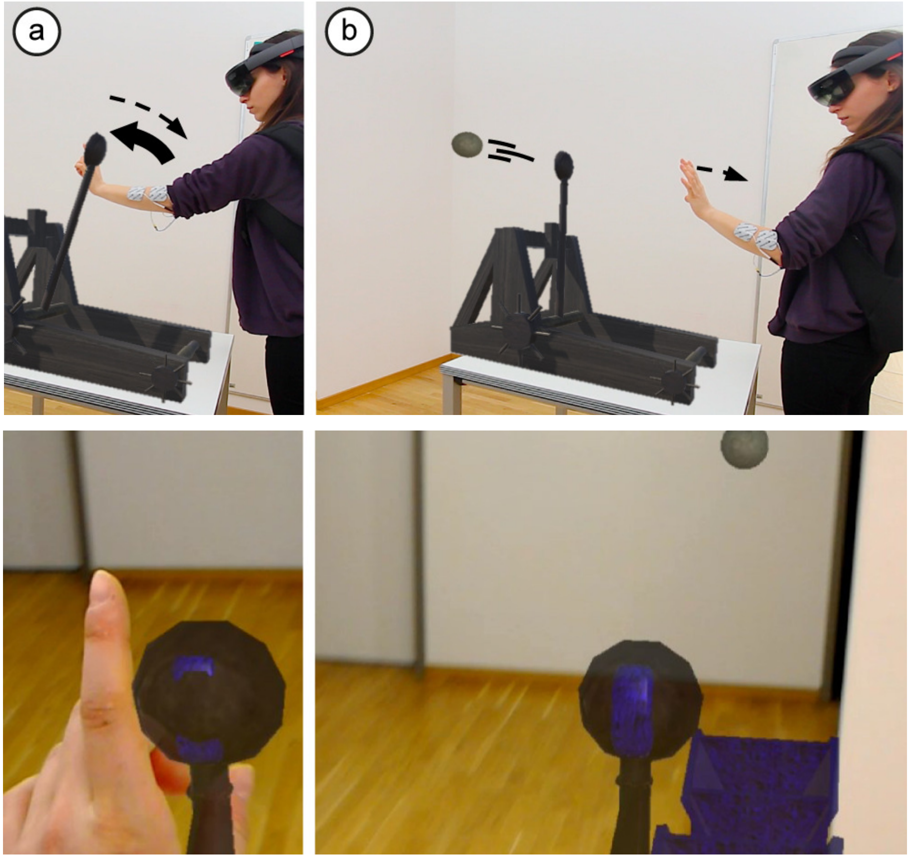

Figure 61: While the user pulls the lever of this virtual catapult, our system provides force feedback simulating the catapult's spring.

2. Augmented physical props: balance marble game. Figure 52 depicted a classic MR marble maze, which was inspired by that of Ohan et. al [116] and complemented with EMS-based force feedback. Here, our marble game renders the shift in gravity caused by the moving marble by actuating the user's arms towards the heavier side. Note that even if the marble is perfectly balanced on the center of the tray, we still render a constant pull down of the user's triceps to represent the marble's weight under gravity.

Furthermore, we render a number of extra haptic effects: (1) when the ball collides with any of the maze walls, we render a short bump in the user's triceps; (2) when the marble falls out of the tray in one of the openings in the obstacle walls, the user feels the relief in the weight; and, (3) when the game starts, the marble falls onto the tray and we render this "weight" by quickly pulling down the user's arms, as depicted in Figure 11.

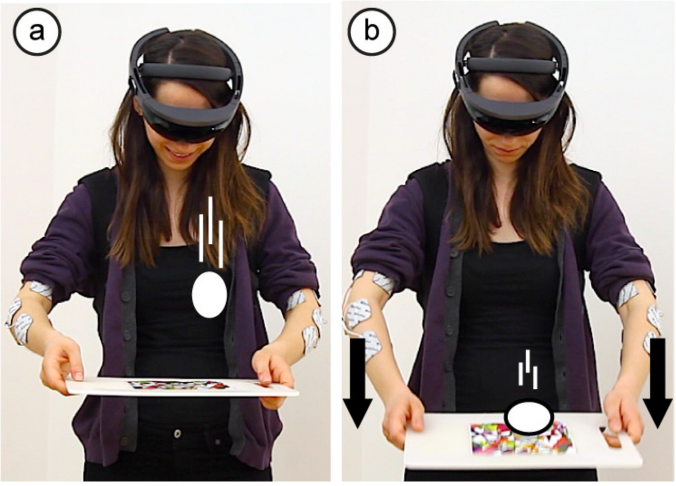

Figure 62: (a) At the start of the game the marble falls from the sky. (b) As it hits the tray, the EMS pulls the user's arms down quickly so as to represent the impact.

3. Augmented appliances: escape room experience. We implemented a simple "Escape Room" experience in mixed reality. In the traditional version of these experiences, users must first find the solution to a puzzle, and only then they can escape the current room. Figure 63 illustrates the user solving this room's puzzle.

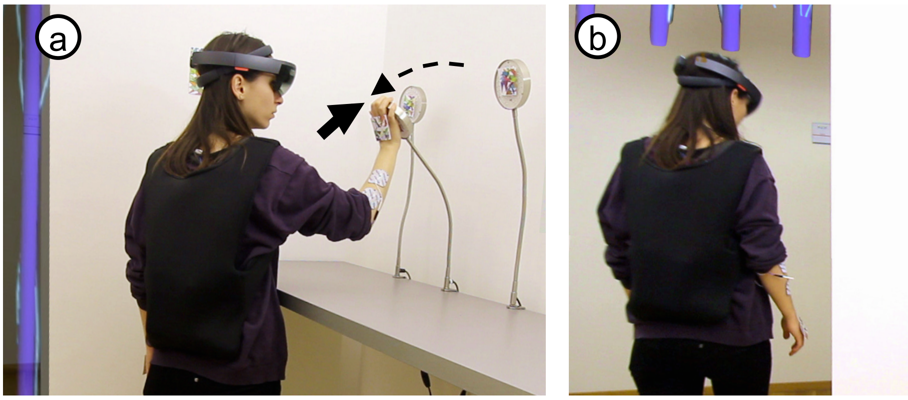

Figure 63: (a) These gooseneck lamps are repurposed as levers, with force feedback, allowing the user to input the secret combination to (b) unlock the door.

The user sees a virtual message on the wall next to the virtual door that provides a clue to the puzzle: “the lights will illuminate your path, but only in one special way”. The user now explores one of the three suspicious gooseneck lamps in the room.

As shown in Figure 63, the user finds that, when moved, the lamps have detents, rendered using our system; hence, they can be only in one out of three positions. By testing different positions, the user finds the secret combination of the lamps’ gooseneck positions that open the virtual door.

3.4.4 Contributions, benefits, and limitations

We propose the use of EMS for force feedback in mixed reality. The main benefit of our approach is that it leaves users’ hands free, thus allowing users to interact unencumbered—not only with virtual objects—but also with *physical* objects in the user’s surrounding, such as props and appliances. We demonstrate this approach by sampling it at several points across the reality-virtuality continuum.

The benefits of our system, which have been validated in our user study, are: (1) providing force feedback to MR leads to a better understanding of the virtual object’s state. (2) This haptic information is especially useful because current headsets have a limited field of view. (3) The addition of force feedback to MR games and experiences increases the perceived realism.

Our system is subject to the limitations of EMS systems: (1) it requires electrode placement and per-user calibration prior to use, (2) it can cause muscle fatigue, (3) and the actuation of hands is typically limited to a single dimension of translation. Also, in order to keep our rendered haptic effects robust to work for all our participants, we opted for fairly simple output gestures (extension of wrist, rotation of wrist, and so forth). Furthermore, our current implementation based on HoloLens has a limited field of view and requires markers to track the orientation of the users’ hands and physical objects. Lastly, while our approach is the first step towards mobile and unencumbered force-feedback in MR, its current form lacks integration with tactile feedback on the fingertips, which we discuss later.

3.4.5 Implementation

To assist readers in replicating our design, we now provide the necessary technical details and the complete source code. The latter allows the community to build EMS enabled applications, which are decoupled from the headset technology by providing an EMS library for Unity3D.

3.4.5.1 Hardware

We implemented our system using a Microsoft HoloLens MR headset [65] and a medical-grade muscle stimulator shown in Figure 64. The interface between these components is a laptop computer running Windows 10 that the user carries in a slim backpack. The laptop is required only to offer a USB to connect the muscle stimulator (so headsets with USB connectivity would require no laptop).

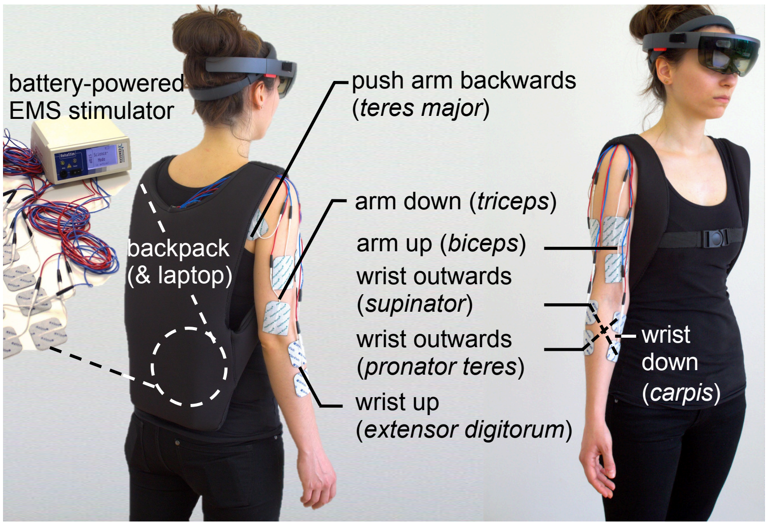

Figure 64: The hardware components and electrode placement (one arm only).

3.4.5.2 Electrode placement

Figure 64 details how 10 electrodes are placed on the user's right arm and shoulder; the user's left arm is equipped the same way. This set-up allows our system to achieve the following motions: (1) wrist extension (*extensor digitorum*), (2) wrist flexion (*flexor carpi radialis*), (3) wrist pronation (*pronator teres*, achieved with the base electrode of wrist extension and end-point of wrist flexion), (4) wrist supination (*supinator*, achieved using both base electrodes of the wrist extensor and flexor), (5) elbow flexion (*biceps*), (6) elbow extension (*triceps*) and (7) pushing the arm backwards via the shoulder (*teres major*).

3.4.5.3 Calibration

To calibrate the system, we (1) start from zero and increase the amplitude of the EMS until we observe a small movement of the muscle. (2) We ask the users themselves to slowly increase the amplitude up to an upper bound that is still comfortable and pain-free while clearly performing the expected gesture. (3) To test the calibrated gesture under realistic conditions, we ask the user to move the arm and we apply the calibrated gesture mid-way. Then, if needed, we re-adjust the intensity to match the expected outcome. This helps our system perform well despite varying arm poses, which tend to occur during real walking. (4) Then, we upload these intensity values to our system. (5) We repeat this procedure for all muscles stimulated in our MR games and experiences.

3.4.5.4 Hand tracking

In our system, the primary way of tracking users' hands is through the HoloLens' built-in camera tracking. When the HoloLens recognizes the point gesture (fist closed and index pointing up) and it reports the 3D position of the center of the hand to our application. Our application applies this position to a representation of the hand (a box collider), which is also the boundary we test for collisions with other virtual entities.

Several of our application examples, e.g., catapult or pushing furniture, run based on the HoloLens' tracking alone. However, for other applications we added *Vuforia* passive AR markers [71] to the back of the user's hands, as shown, e.g., in Figure 54b. These help us overcome two current issues with the HoloLens' hand tracking (Version 1, Development Kit 2016): (1) hand tracking fails when the user's hand comes close to a real-world surface and (2) the HoloLens API reports just the position of the hand, but no orientation.

3.4.5.5 Mapping virtual to physical space

In order to attach virtual contents to locations in the physical room, our system uses HoloLens *spatial mapping* feature [65]. Our system also utilizes the *HoloToolkit-Unity* library alongside its *Spatial Understanding* module during runtime to identify surfaces where virtual objects may be placed.

3.4.5.6 Event handling

Our system processes events as follows. (1) When, running on the HoloLens, the system detects a collision between the user's hand and a virtual object by means of a Unity *Collider*, it (2) determines the parameters of the muscle stimulation based on the physics properties of the virtual object (weight, friction coefficient, springs) that the user is interacting with. Pushing a couch, for example, produces a stronger haptic effect than pushing a button. (3) Then, the system generates the message for the EMS stimulator and (4) sends it via UDP to the laptop. (5) The laptop, which is running a simple Unity3D UDP server, receives the message and (6) forwards it in a serial format (via USB) to the muscle stimulator. (7) This triggers the stimulation.

3.4.5.7 Technical limitations of our prototypes

While, to our knowledge this is the first functional mobile implementation of EMS in Mixed Reality, there are a number of limitations in the current prototype.

Given that the HoloLens does not provide a USB port, there is some latency in the wireless implementation: detecting collisions in Unity (~20-30fps), sending EMS commands wirelessly to the stimulator (<10ms) and detecting the hand's position via Vuforia (~20fps). Note, however, that our participants did not remark on perceived latency while commenting on their experiences.

On the haptics side our prototype is limited to force-feedback, allowing users to feel the boundaries of objects as well as forces arising from interactions, but does not render any textures on the fingertips. However, it does render some basic physics of objects (even beyond just soft vs. hard as in [122]), for instance: the force of the spring inside the button, the couch's static friction being larger than its kinematic friction, the couch being softer in middle than at the arm rest and so forth.

Lastly, our games are currently based on the HoloLens room tracking, i.e., they require the HoloLens to recognize the current room [65]. Within a room, the full interaction area is available for EMS feedback as long as users' hands are inside the HoloLens front-facing hand tracking area.

3.4.6 User study

The objective of our experiment was to assess whether EMS-based force feedback in MR increases the users' sense of realism.

Participants performed three simple tasks, directly derived from our aforementioned applications, each task using an EMS-based condition and a no-EMS control condition. We hypothesized that EMS would lead to higher ratings on both realism and enjoyment.

At this initial stage of exploring EMS in MR games, we opted for a study focused on the realism of the proposed haptic effects. In order to do so, participants were instructed to explore the EMS-induced physical sensations rather than to perform the task at maximum speed. As such, our study does not provide insights into task performance.

3.4.6.1 Apparatus

Participants wore the HoloLens and our EMS backpack-based setup discussed earlier and depicted in Figure 64, which allowed for untethered use. We calibrated the EMS setup as described earlier (see "calibration"). Participants experienced the sound effects of each task via the HoloLens headphones.

3.4.6.2 Interface conditions

Participants experienced two distinct interface conditions: (1) *EMS*: force feedback by means of EMS. (2) *no-EMS*: participants received no force feedback (control).

3.4.6.3 Tasks

1. **Furniture**: Participants performed a simplified version of our walkthrough in which they were asked to rearrange a virtual lamp on the table to properly light a physical book, and align a virtual couch with the room's wall.
2. **Catapult:** Using our catapult, participants tried to hit two targets at different distances.
3. **Marble:** Participants played the marble balance game in three levels of increasing difficulty.

To the study concise and under one hour we excluded the escape room experience.

3.4.6.4 Procedure

Each participant performed all three tasks in both EMS and no-EMS condition (within-subject design). Condition and task order were randomized. After each task, participants rated their experience in terms of *realism* (1: artificial, 7: realistic) and *enjoyment* (1: not at all, 7: very much). After all tasks, we asked participants which interface condition they preferred and gave them an opportunity to provide open-ended feedback.

3.4.6.5 Participants

We recruited 12 participants (2 female, $M = 22.7$ years, $SD = 4.9$) from our local institution. Two out of the 12 participants had tried a HoloLens at a technology fair. One participant had previously experienced EMS at a local gym. With their written consent, we videotaped the study sessions and transcribed their comments.

3.4.6.6 Results

We analyzed the data using a 2 (*condition*) × 3 (*task*) repeated measures ANOVA ($\alpha = .05$) as suggested by [113]; all pairwise comparisons were Bonferroni-adjusted.

Figure 65 shows participant's average ratings, in both conditions, regarding perceived realism. We found a significant main effect on interface ($F_{1,11} = 46.112$, $p < .000$). Pairwise comparisons revealed that perceived realism was higher in the EMS condition, for every task.

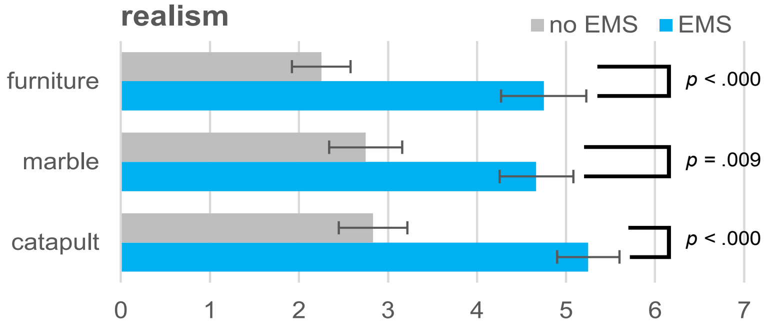

Figure 65: Participants rated their experience as more realistic with EMS.

We also found a main effect of the interfaces on enjoyment ($F_{1,11} = 17.135$, $p = .002$). As depicted in Figure 66, participants rated the enjoyment significantly higher when in the EMS condition for the *furniture* and the *catapult tasks*.

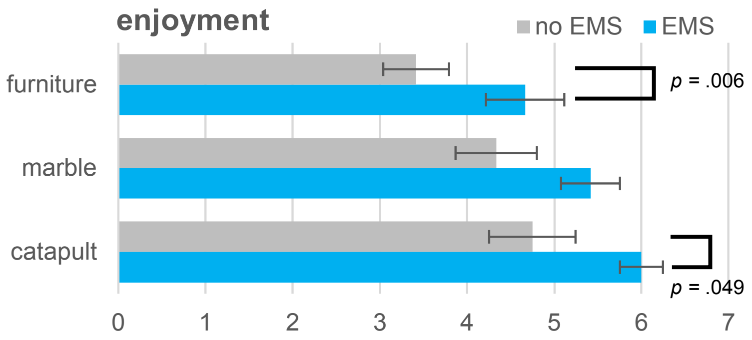

Figure 66: Participants rated their experience as more enjoyable in most of the EMS conditions.

Figure 67 summarizes participants' preferences for each of the interface conditions. For the *furniture* and the *catapult tasks*, 10 out of 12 participants preferred the EMS interface condition, for the *marble task*, 7 out of 12 preferred EMS.

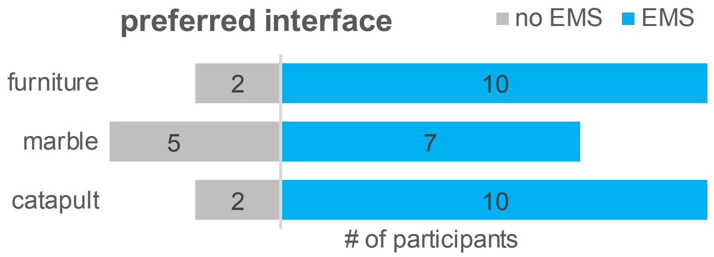

Figure 67: Most participants preferred EMS to no-EMS interface condition.

3.4.6.7 Qualitative results

Throughout the experiment participants often voiced their explanations for what they felt, albeit not being instructed to do so. We summarize these:

Pushing furniture. When pushing the lamp and couch most comments revolved around how the EMS aligned with their expectations on the physics of objects, such as friction, weight, and collisions. For instance, P5: "I immediately feel when I touch something [shows us touching the couch], even though I know this [couch] is not here" (similarly P6, P9, P10). P2 noted "pushing the lamp and couch felt much better than I ever expected, it's like you feel the weight, and the couch was heavier". P5 "did I feel friction? I think I felt friction when pushing it [the lamp] on the table". P3 "was super real to push the lamp, only in the end I realized that when the lamp hit that thing [bump] on the table, I had to push stronger against the muscle stimulation to make it pass it" (similarly, P9, P10 and P12). Furthermore, seven out of 12 participants remarked how, in their opinion, EMS had helped them overcome the HoloLens' limited field of view. For instance, P5: "the muscle feedback makes pushing the couch much easier, (…) I could not see if it was hitting the wall, but I could feel it".

Balancing the marble. Using the tray (Figure 68) polarized participants in that only 7 of them expressed a preference for experiencing it with EMS. Not remarkably, 8 participants commented that the EMS added difficulty to the game, because not only they had to steer the marble, but also they had to do it against the force feedback. P8 remarked, "[EMS] makes it feel like a heavy ball when it pushes me down (…) but when it pushes me to the side it makes the game harder and more confusing". In contrast, P5 commented "this [EMS] makes the marble motion realistic, [and thus] the game is now much nicer to play".

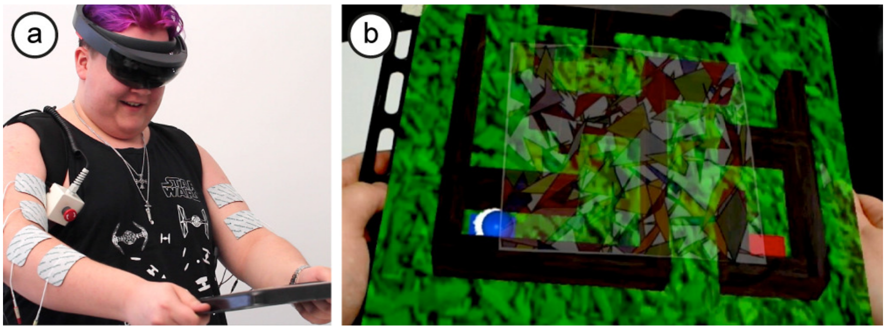

Figure 68: Participant balancing the marble (with consent of the participant).

Catapult. All participants voiced positive remarks about the catapult, which might explain the highest rating for both realism and enjoyment. P9 noted "Ah, now I feel the catapult's arm", when trying the catapult with EMS after trying without. Similarly, P2 noted "it helps me to know how far I will shoot, because I feel the amount of [EMS] feedback is related to how much I pull". P1 noted "without EMS the catapult is no fun because it becomes a memory game, you learn the correct angle and just hit the targets". P8 stated "this is how I think pulling a catapult feels like".

Open-ended feedback. P3, P5, P6 and P7 expressed their appreciation for the physical objects we had integrated into the MR experience, e.g., "I like touching these real things [points at the book on the table, the wall and the tray] and feeling they are now part of the VR" (P7). When we asked participants what would be required to reach the level of realism they would expect, participants' comments included "adding tactile stimulation to my hands", "larger field of view", and "finer EMS motions for the marble game".

3.4.7 Discussion of our findings

Our study results support our hypothesis, i.e., *EMS* did indeed significantly add realism to the three mixed reality tasks. And, for two out of the three tasks, EMS had also significantly increased participants' enjoyment. These findings are further supported by participants' generally positive comments including "I like touching these real things [pointing at the book on the table]", which is the essence of why we used EMS to implement this functionality while keeping the users hands free.

At first glance, our findings are aligned with our EMS research in VR, e.g., impacto and VR walls. However, we also observed that, unlike in VR, exaggerated haptic effects (e.g., the impact of the ball when falling on the tray was bigger than of a ball that size) fall short of illuding the user in MR. As P3 noted “but this marble that I see cannot possibly be that heavy, I can see the world around me, so I can imagine the weights [of things]”. In fact, P3 was pointing out a core quality of MR, which arguably makes it different from VR. In MR, experiences take place in the context of the physical world and thus users have a keener sense for plausibility. In the case of MR, we observed users remarking how they enjoyed nuanced aspects of the EMS-enabled physics, for instance: “I can feel the couch is harder to move when it is stopped [due to our EMS-based static friction]” (P3). As a recommendation for UX designers working in MR, we suggest aligning the “haptic-physics” with the expected physics as much as possible rather than resorting to exaggerations.

Lastly, it is worth noting that our study is limited in that it deals with the realism of the EMS haptic effects in MR games; hence, we asked users to freely explore the task at hand. Therefore, these findings cannot be generalized for other tasks that require a measure of performance or precision.

3.4.8 Conclusions on EMS as mixed reality force feedback

We demonstrated a fully mobile system that empowers mixed reality games experiences with mid-air force feedback by means of electrical muscle stimulation. The main benefit of our approach is that it leaves users’ hands free, thus allowing users to interact unencumbered with *physical* objects in their surroundings, such as props and appliances.

Besides the direct implications for increased realism in MR gaming, EMS might uncover new terrains for augmented passive objects and appliances. For instance, an appliance that is augmented with EMS might have more potential if we think of using it daily. Unlike *RetroFab* [129], which complements the appliance with updated hardware UI, an EMS-augmented appliance allows updating the UI of a device by merely updating the *software* (i.e., the EMS side). Also, our tangible dial that automatically recalled the last position, i.e., the cup in the “walkthrough”, points to another strength of exploring EMS in MR: EMS might assist in aligning virtual and physical realities to prevent inconsistent states often introduced by physical props (as debated in [98, 72]). While previous methods solved this by mechanically coupling or actuating the props (e.g., mechanically constrained tangible dials [72]), EMS allows for everyday handheld objects to move without instrumentation.

We see this research as a first step towards more physical mixed realities by adding force feedback. The next steps might include combining this approach with tactile feedback, especially techniques such as back-of-the-nail vibrotactile [3, 4], which does not occlude the fingertips.

3.5 EMS & mechanical-actuation: sides of the same coin?

Let us draw attention to the common theme running throughout this chapter: the apparent similarities between EMS-based and mechanical solutions. It was precisely this observation that led us to propose EMS as an alternative to motor-based haptic devices. But now, with a critical perspective, we must ask ourselves: how far do these similarities go? Might the two technologies even be interchangeable?

3.5.1 Similarities between EMS and mechanical actuation

There are certainly a number of apparent similarities. For several applications, we can find analogous implementations based on mechanical actuation as well as based on EMS. As we have described, Gesture Output [132] actuates the user's fingers using a clear touch screen overlay pulled by motors, while our *muscle-plotter* device (which we will introduce in Chapter 4) achieves a similar effect using EMS. Likewise, *Third Hand* [97] uses a forearm-mounted robotic arm to provide force-feedback to a mobile phone, while our muscle-propelled force feedback achieves this effect based on EMS.

As illustrated by Figure 69, many of the application scenarios today explored using EMS, which we covered extensively in Chapter 2, had already been explored using mechanical actuators earlier, at least to some extent. Teleoperation—precisely transmitting forces between two remote users—has been a flagship area for mechanical actuators since the early days of robotics/haptics. This is due to the precision and speed provided by systems based on mechanical actuation. In rehabilitation, exoskeletons are used for gait regeneration. Also, many high-end training and tutorial systems use haptics to enhance learning of physical tasks such as palpation training. Likewise, early force feedback devices were constructed to increase the realism of virtual experiences.

	Mechanical	**EMS**
Teleoperation	1948 [43]	1995 (art [43])
Replacing motor functions	1969 [106]	1979 [145]
Information Access	1973 [136]	2015 (pose-IO)
Tutorial / Training	1980 [133]	2011 [146]
Immersion	1992 [14]	2006 [34]

Figure 69: EMS application areas follow research areas in mechanical actuators.

Given that most of these applications areas are analogous, one starts to wonder to what extent the technologies might be interchangeable, i.e., if an application was previously implemented using a mechanical-based actuator, can it then be replicated by means of EMS? The short answer is: not necessarily.

3.5.2 Advantages of mechanical actuation

One factor is that EMS-based systems can never be more powerful than a human, while mechanical systems can. In fact, current EMS-based actuation tends to be substantially weaker than voluntary muscle contraction, because EMS systems tend to recruit only a subset of the motor fibers that can be recruited by voluntary actuation. This prevents some applications, such as super-human exoskeletons, from ever being translated into EMS-based systems.

While output force is certainly one of the limitations of EMS, a much bigger limitation is precision. As of today, EMS-based systems are less precise than mechanical systems and, therefore, are usually employed for generating simple poses such as the wrist motions presented above; this is, in part, due to the layered nature of the human muscles: since electrodes are attached to the skin, they cannot target a specific muscle without affecting nearby muscle tissues. Furthermore, EMS actuation competes with the user's voluntary muscle activity. This has implications on applications, i.e., those applications that require precision and reliability, such as robotically assisted surgery, can today only be handled using mechanical actuators, not EMS.

Based on the discussion above, one may wonder whether mechanical actuation is just generally the more promising approach but, again, the answer is: not necessarily.

3.5.3 Advantages of EMS-based actuation

A key strength of EMS, which we already discussed earlier, is that EMS systems miniaturize well and, therefore, are suited for mobile and wearable applications. EMS-based systems are generally lighter and require less energy than mechanical actuators.

Another interesting quality of EMS systems is that they reach more actuation sites than mechanical actuators. Whereas mechanical systems require at least two mounting points, for an EMS-based system attaching a pair of electrodes to the target muscle is sufficient—which allows EMS to also be applied to the torso muscles (e.g., abdomen, lower back, diaphragm, etc.). For instance, *Vibr-o-matic* (covered in Chapter 2) stimulates muscles of the abdomen and larynx to allow novice singers users to sing in vibrato [38]. EMS has also been used to simulate food texture by actuating the jaw muscles or even assist patients in swallowing. Any of these actuation sites would hardly be reachable or practical with a mechanical actuator.

So, in conclusion, at this stage EMS and mechanical actuation both have their strengths and limitations. Mechanical actuation is clearly advantageous when it comes to power and precision, while EMS is best used when it comes to wearable form-factors and actuating areas such as the torso.

4 INFORMATION ACCESS USING EMS

While our first line of work explored EMS as an alternative to increasing realism that departed from traditional force feedback devices, in this chapter we turn our attention to a different angle: *using EMS to access information via poses of the body*.

This line of work echoes our initial remark that the evolution of computing shows devices and users getting closer to each other. Most recently, we witnessed how mobile smartphones and wearable devices became the de facto way we access information [13, 17]. Wearables are special in that they bring the device into constant physical contact with the user's skin and allow users to interact with information anywhere, anytime. Most wearable devices are designed to minimize interference with the user's primary task, such as walking or having a conversation [7]. In fact, many wearable devices are designed specifically for eyes-free use by instead relying on auditory feedback [10, 135].

Unfortunately, using audio as an output modality interferes with auditory tasks, such as having a conversation, and limits interactions to quiet environments. Haptic interfaces, such as vibrating bracelets, show promise here [119], but are limited in terms of bandwidth [109] and are hard to learn because they lack mnemonic properties [94].

We therefore proposed a different type of modality for wearable interaction, namely *proprioception*, which is the users' sense of the relative position of neighboring limbs of the body. As we demonstrate in this chapter, resorting to three different projects we have created, we can use proprioception for both input *and output*.

4.1 Proprioceptive Interaction: poses for input and output

Figure 70 illustrates the concept of proprioceptive interaction. Proprioceptive interaction allows for input and output by posing one of the user's limbs, here the user's wrist. Users enter information by performing an input gesture, here flexing their wrists inwards, which the device senses this using its accelerometer. Users can perform such a gesture eyes-free, as their proprioceptive sense informs them about the position of their wrists. Users receive output from a proprioceptive interface by finding their body posed in an output gesture, here again the wrist. Also, they perceive it eyes-free by means of their proprioceptive sense.

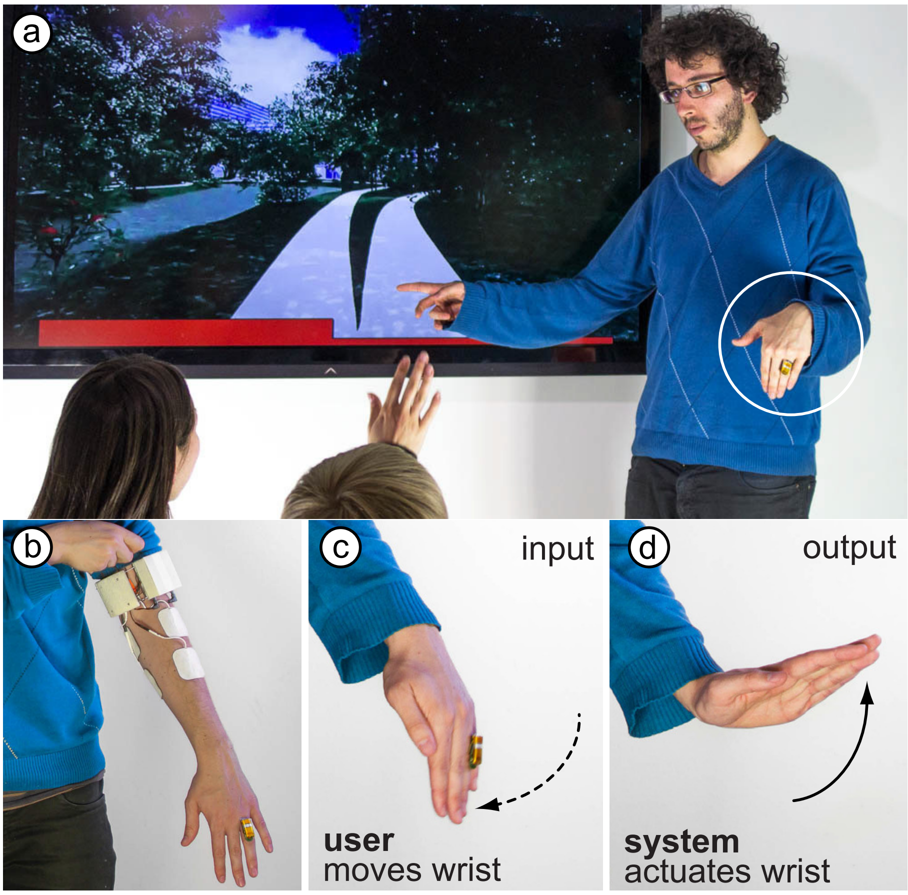

Figure 70: Using the sense of proprioception alone to rewind a video, while maintaining eye contact with his audience.

The interaction shown in Figure 71 is of the "purest" form in that input and output occur through the same limb, here the wrist. This "symmetric" interaction results in a particularly intuitive interaction [132].

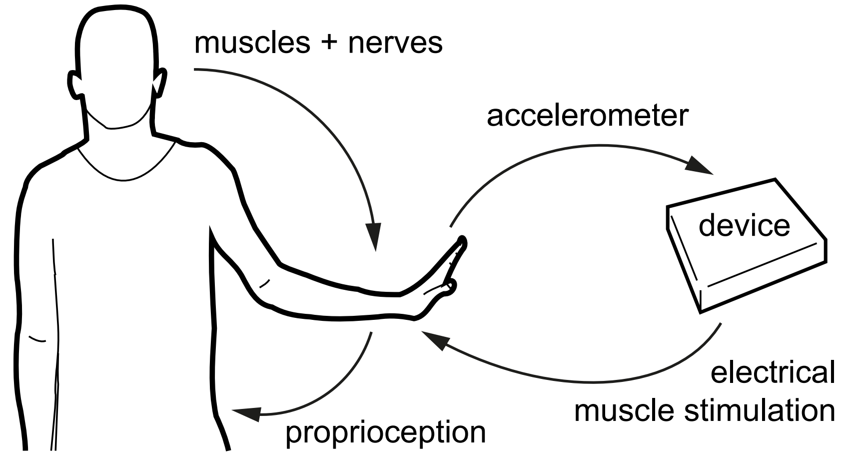

Figure 71: Symmetric proprioceptive interaction revolves around the pose of one of the user's limbs, here the wrist.

However, proprioceptive input and output may also occur through different limbs, e.g., when the application requires more input than output or more output than input. We call this asymmetric proprioceptive interaction, which is depicted in Figure 72.

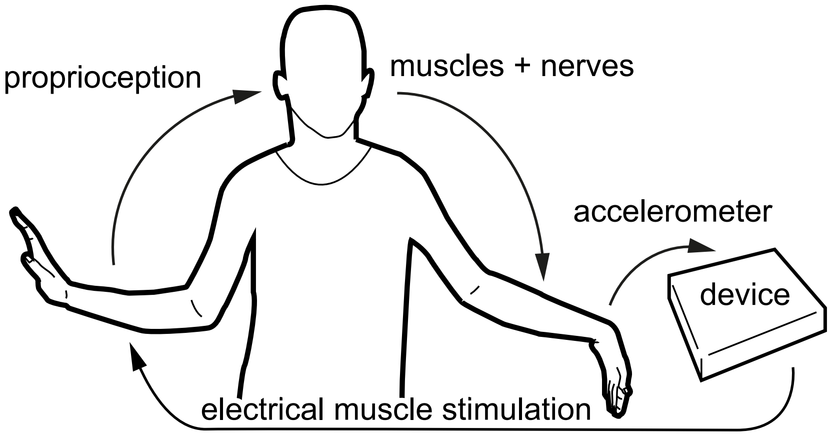

Figure 72: In an asymmetric proprioceptive interaction, input and output occur through different limbs.

4.1.1 The pose-IO device

Figure 73 shows pose-IO, the proof-of-concept proprioceptive device we have created. It offers input and output based on proprioception. We implemented pose-IO in the form of a bracelet, which users wear under their clothing. Pose-IO receives input from the user by means of a wireless accelerometer attached to the user's hand. Pose-IO produces output by writing to the user's arm muscles using electrical muscle stimulation (EMS) through four electrodes attached to the user's arm.

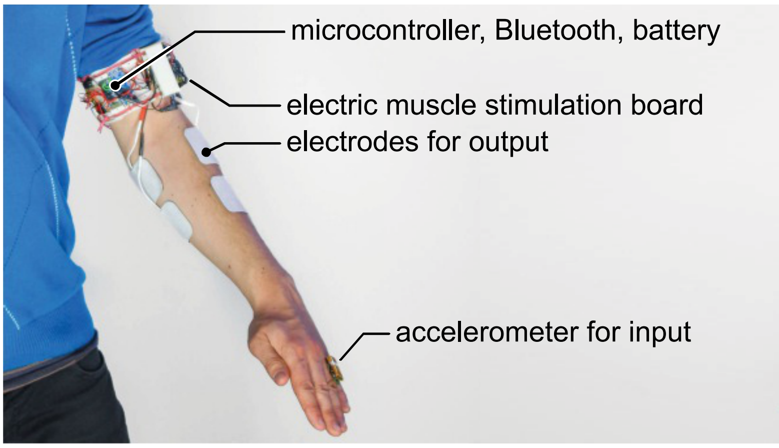

Figure 73: The main components of pose-IO.

4.1.2 Applications

We have implemented three simple applications, each of which highlights a different strength of proprioceptive interaction.

4.1.2.1 Symmetric proprioceptive interaction

Figure 70a illustrates our video-scrubbing tool for presenters based on the pose-IO device, which is an example of symmetric proprioceptive interaction. One of the audience members just asked whether the presenter could show the video one more time; Figure 74 shows how the presenter responds.

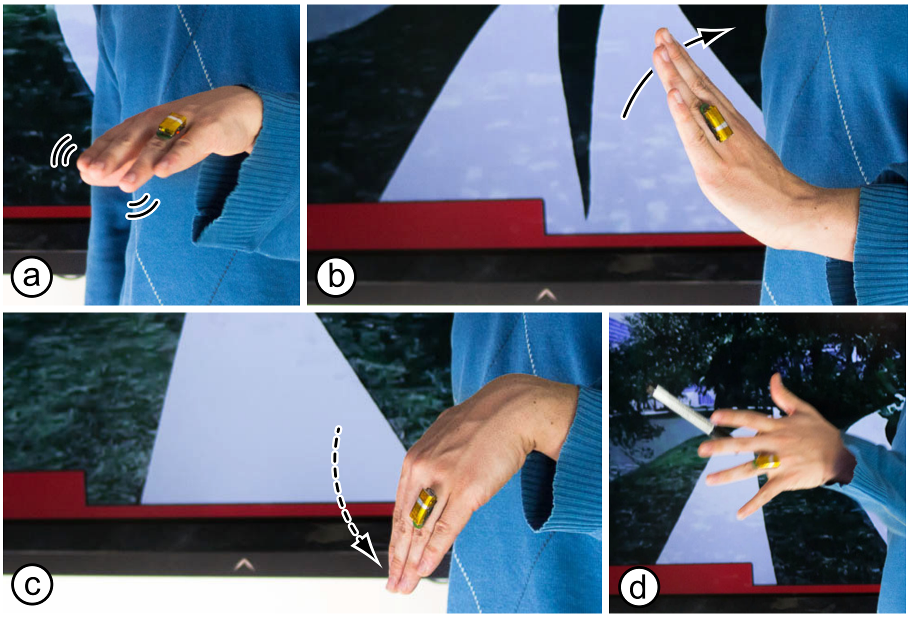

Figure 74: Using pose-IO to scrub a video, i.e., one’s wrist as the I/O device.

First, (a) he invokes pose-IO's video scrubber by performing an invocation gesture, here by shaking his wrist back and forth. (b) pose-IO responds by posing the users' wrist, so as to represent the current video playback position, which the user perceives through his sense of proprioception. The user lets the video run for another moment during which pose-IO keeps updating the user's wrist position, continuously extending it upwards, which again the user perceives through his sense of proprioception. (c) The user now rewinds the video by flexing his wrist back down to the desired play position. He uses sufficient force to override pose-IO's control over the wrist. Pose-IO responds by scrubbing in real time, allowing the user to locate the intended position.

(b, c again) To emphasize his answer, the presenter replays the scene one more time. This time, he moves the video play head to its intended position without iterating, but by posing his wrist directly in the same position as before, based on his proprioceptive sense alone. (d) The presenter is just about to dismiss pose-IO using a strong flick gesture, when one of the attendees throws him a marker. He catches the marker and pose-IO interprets the abrupt movement as a dismiss gesture and deactivates itself. This is by design, so as to allow users' primary tasks to take priority whenever necessary.

This particular scenario uses the same limb for input and output, allowing us to implement a *symmetric* interaction, i.e., an interaction where the same modality and the same mapping are used for input and output, resulting in a particularly intuitive interaction that even allows users to learn input based on output [132]. This is a benefit of proprioceptive interaction over other wearable interfaces, for example, such as vibrotactile feedback, which has to be combined with a different input modality, such as a set of buttons.

4.1.2.2 Asymmetric interaction

Proprioceptive interaction does not have to be symmetric though. We demonstrate this at the example of the eyes-free reaction game *red hands* [52]. Traditionally, this children's game is played by two players, one of which is trying to slap the hands of the other. With the use of pose-IO, we turn red hands into a solitaire computer game. We partition the user's body as depicted in Figure 75, which is why the interface is asymmetric.

Here the user's right hand represents the computer opponent; the left hand continues to belong to the player. At every turn, the computer hand tries to slap the player's hand (i.e., game *output*); the player's objective is to evade the slap (i.e., game *input*). The red-hands application detects successful slaps based on *acoustic* recognition (similar to [54]), i.e., a wearable microphone in a small bracelet around the player's wrist.

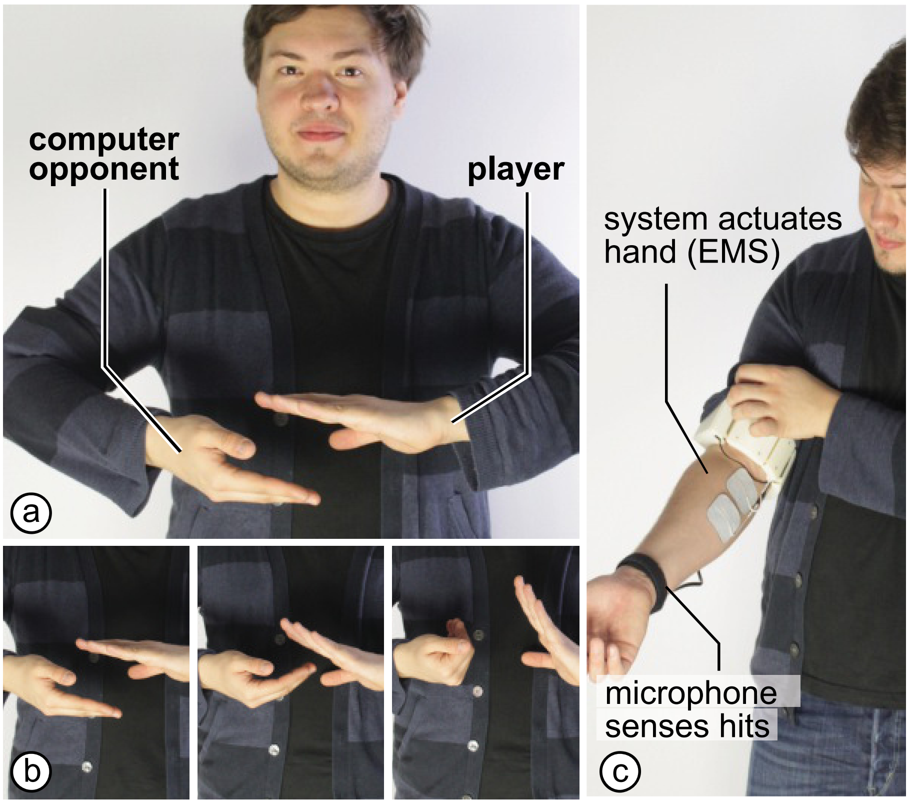

Figure 75: Red-hands for one: under control of pose-IO, the user's right hand tries to slap the user's left hand. The goal is to evade getting slapped by his own hand.

Red hands is an example of asymmetric proprioceptive interaction in that output goes to the right hand, while the device receives input from the left hand.

To make red hands playable, it offers multiple levels of difficulty. The first level gives players several hundred milliseconds of advance warning before slapping. It does so by briefly flicking the computer hand before actually attacking. This warning period gets shorter on subsequent levels. Since pose-IO talks directly to the user's muscles, it can actuate the hands surprisingly fast: if we set the warning period to zero, we found player's chance to evade the slap to be no better than random (see also Study 2). This ability for *very* fast interaction stems from talking to the user's muscles directly—a key strength of our approach.

Similar to the presenter tool, red-hands for one is played through the sense of proprioception. This allows playing eyes-free, which, interestingly, most of our study participants did even though they were allowed to look (see Study 2 for details).

4.1.2.3 Asymmetric interaction with a spatial component

To demonstrate an interaction with a spatial component, we implemented *imaginary pong for one.* As depicted in Figure 76, we again partition the user's body so that the actuated hand forms the computer opponent, while the other belongs to the player.

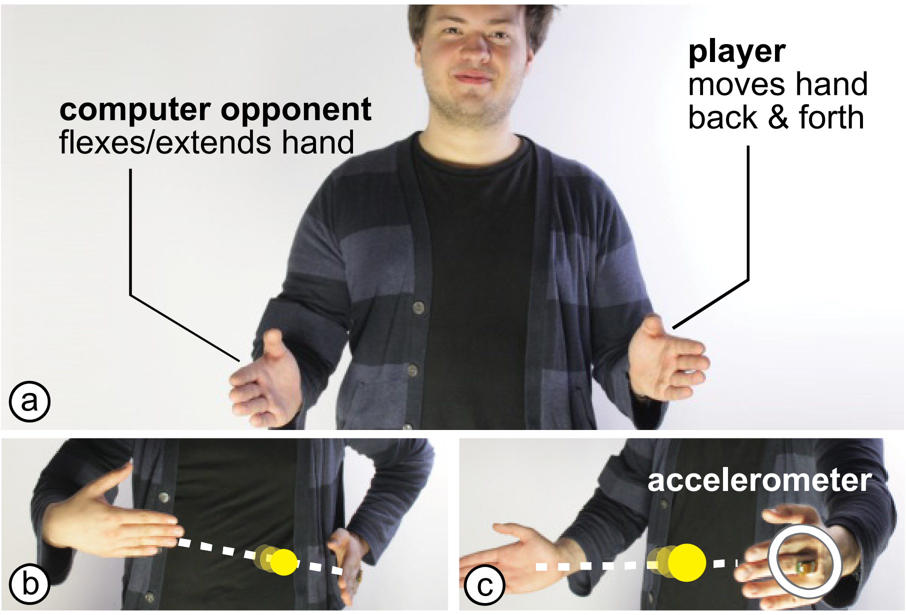

Figure 76: Imaginary pong for one: (b) The computer opponent plays the imaginary ball left by flexing its hand (c) The user returns the shots by moving his hand to where he feels the ball went.

There is no actual rendition of a ball. Instead, players have a notion where the ball should be based on their sense of physics (similar to [11]). The computer opponent may flex the user's hand to play the imaginary ball towards the player's body or it may extend the hand to play the ball away from the user's body. The user's task is to return the shots by moving his hand to where he feels the ball went. In one version of the game, we added a second pose-IO bracelet to the player hand in order to render recoil at the moment the imaginary ball hits his palm. This allowed us to add a timing element to the game. As with the two applications presented earlier, users play eyes-free through the sense of their proprioception alone.

4.1.3 Contributions, benefits, and limitations

Our main contribution is the concept of proprioceptive interaction, here illustrated at the example of wrist pose. In addition to being wearable, proprioceptive interaction offers four desirable properties: (1) It allows for eyes-free and ears-free use. (2) Instant invocation and dismissal allows users to invoke the device any time and also to return to their primary task immediately when necessary. (3) Symmetric setup allows implementing the same interaction "language" for input and output, resulting in a particularly intuitive interaction that even allows users to learn input based on output [132]. However, (4) this I/O symmetry is not mandatory, and we also explore an interaction modality in which input and output occur at different limbs; we denote it as asymmetric interaction. Thus, depending on the type of application, designers might place input and output freely to best fit their needs.

On the flipside, interaction based on the perception of poses is obviously very low in bandwidth compared to visual or auditory interaction. Its use is therefore only applicable for very simple tasks. Also, any solution that relies on EMS requires an initial step in which the system is calibrated.

4.1.4 Implementation details

To help readers replicate our device, we now provide a detailed description of it. As shown in Figure 77, pose-IO's hardware design has the form of a 3D printed bracelet with attached electrodes.

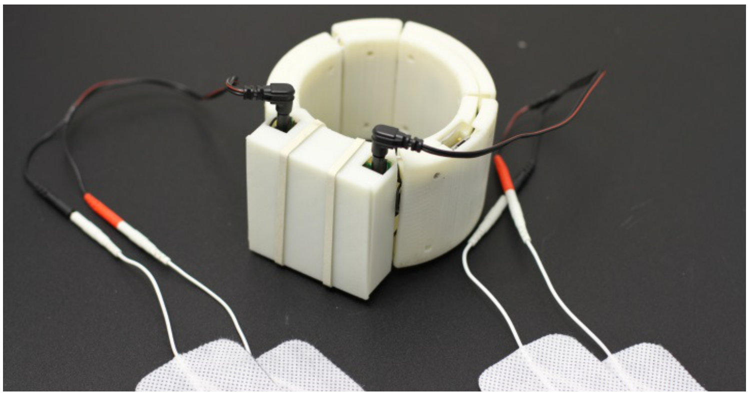

Figure 77: The 3D printed pose-IO bracelet.

Figure 78 shows the opened-up bracelet. Pose-IO writes output to the muscles using a medically compliant electrical muscle stimulation unit (TruTens V3) connected to four pre-gelled electrodes (50 x 50 mm). The amplification for the Electrical Muscle Stimulation is regulated by two CMOS digital potentiometers with non-volatile memory (X9C103, 10 KΩ) controlled by an Arduino Nano using a three-wire serial interface.

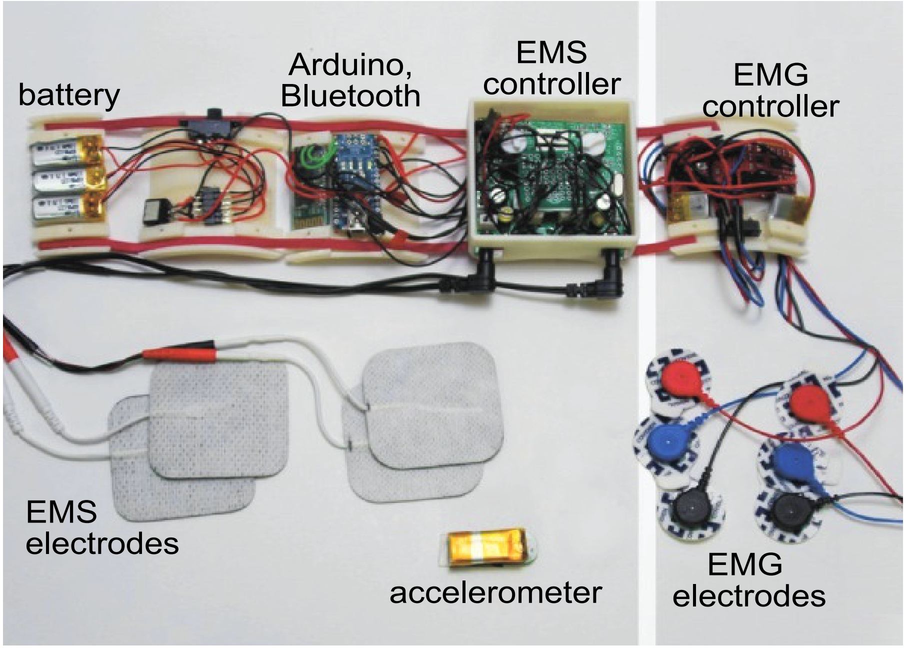

Figure 78: The pose-IO bracelet unrolled and with the top enclosure removed.

4.1.4.1 Electrical muscle stimulation parameters in pose-IO

Pose-IO creates its EMS signal as a biphasic waveform. Its signal pulsates at 120 Hz with a pulse-width of 150 µs. The current is limited to 100mA, allowing for safe operation. Given that the lowest power settings of the EMS unit did not achieve muscle actuation with the users we tested with, we stepped up the control curve by adding a 10 KΩ resistor in parallel to the EMS unit's variable output and input pins. This allows pose-IO to achieve a smoother output current curve.

Upon first use, users calibrate pose-IO's EMS device by specifying the lowest stimulus that still leads to a recognizable sensation (no visible hand motion at this level) as well as the maximum signal that the user perceives as comfortable. At all times, pose-IO use is pain-free. Figure 79 shows the placement of the EMS electrodes on the *extensor digitorum* (wrist extension) and on the *flexor digitorum superficialis* (wrist flexion).

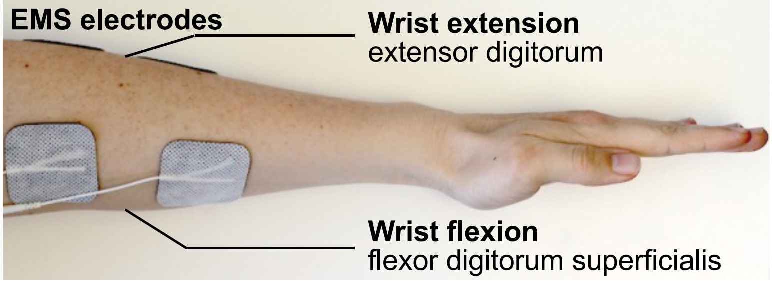

Figure 79: EMS electrode placement for wrist flexion and extension.

4.1.4.2 Accelerometer

Pose-IO uses the accelerometer's Y-axis to determine the user's wrist position as the tilt of the user's hand against the horizontal, sampled at 50 Hz. We also made an extended version of the device that determines the wrist pose with respect to a second IMU worn on the forearm. This allows users to operate pose-IO in any body posture.

Users invoke and dismiss pose-IO by holding the hand horizontally and then shaking it, which pose-IO senses by looking for acceleration along any of the other accelerometer axes. Pose-IO sends commands to its applications, such as aforementioned games, which it runs on the Arduino microcontroller. Pose-IO talks to software running on other computers, such as the video player in the presenter tool scenario, via Bluetooth.

Depending on version, pose-IO either uses an Axivity WAX3 or WAX9 3-axis wireless accelerometer that users wear on their ring fingers. After removing them from their casing, both measure 34.5 mm × 16 mm × 15 mm and offer 4 mg resolution [54]. The WAX-3 sends data over IEEE802.15.4 radio, which we convert using a laptop computer with a radio dongle; we resolved this issue with the newer WAX9, which sends data directly to pose-IO's Arduino via Bluetooth.

Pose-IO is powered by a 9 V lithium ion rechargeable battery. Under continuous use, it offers around 4 hours of battery-life on the bracelet and 8 hours on the accelerometer.

4.1.4.3 Control loop for symmetric input & output

When tracking an external signal, such as the video play position, pose-IO actuates the user's muscles using a PID control loop. We obtain oscillation-free behavior using the gain factors: Kp=1.2, Kv=1.1, Ki=0.5 (Figure 80). Pose-IO calculates the error derivative on a moving average over the 10 last measured velocities.

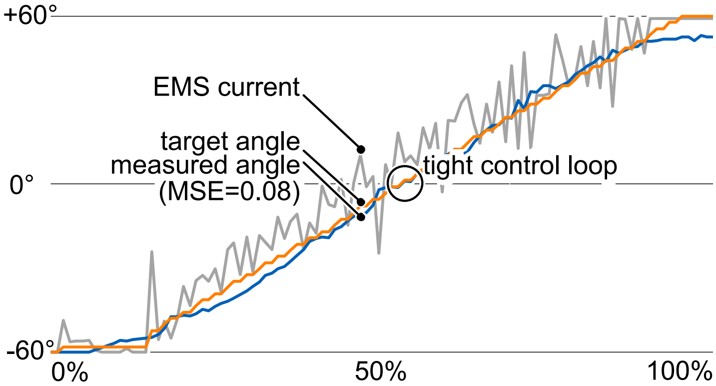

Figure 80: Behavior of pose-IO's PID control.

If the accelerometer value deviates from the system state by 10°, pose-IO assumes that the user intends to override the system state—in the video scrubbing scenario this allows users to override the video play head position.

4.1.4.4 Extended version based on electromyography

The version of pose-IO described so far relies on the accelerometer for input, which users wear visibly on their hand. For applications in which user control and a computer-control are distributed across two hands, e.g., the two games presented earlier, we achieve a fully hidden version of pose-IO by dropping the accelerometer and instead using electromyography (EMG) for sensing. EMG senses muscle activity, which is what the two game applications require. Users invoke the EMG version of pose-IO by over-extending their wrist for 1.5 seconds or longer (Figure 81).

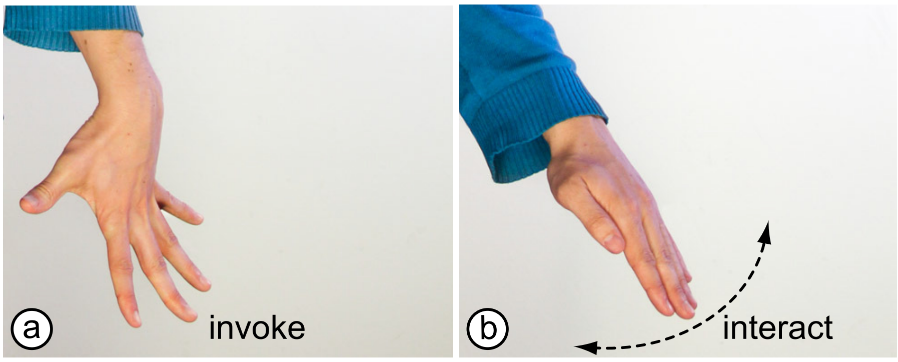

Figure 81: The pose-IO version based on EMG uses hand-overextension as invocation gesture.

The EMG version of pose-IO contains two additional electromyography boards shown on the right of Figure 78; they are based on the AD8221 differential amplifier. The EMG boards are connected to six electrode caps, four to sense and two references (20 mm diameter), which are placed on the same muscles as for EMS, as depicted in Figure 82. The EMG boards are protected from the EMS current using two relays, which cut the EMG pathway if the EMS is active.

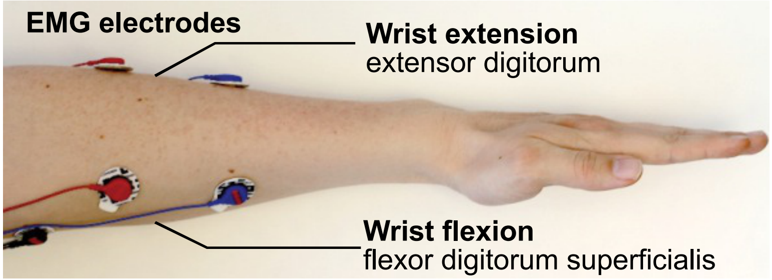

Figure 82: EMG electrode placed on wrist extensor and flexor.

4.1.5 Study of proprioceptive input & output

The purpose of our first study was to verify our basic proprioception interaction concept, i.e., interaction by means of posing the wrist.

The study focuses on symmetric input/output. In the video scrubbing example presented earlier, pose-IO informed users about the play head position by posing their wrists, allowed them to set the play head position by posing their wrist, and allowed them to recall a value by re-creating a known wrist pose. Since the first two have already been studied in the psychophysics literature [154] and [39], we decided to investigate the latter: *re-creating a wrist pose*. This interaction is particularly meaningful in the context of proprioceptive interaction in that it directly investigates symmetric input-to-output correspondence.

In this study, we used the pose-IO device to pose participants' hands. Participants then dismissed this pose and after a pause tried to re-create that pose. Given that participants perceive both poses by means of proprioception, our expectation was to see close correspondence between the two poses.

4.1.5.1 Task

For each trial, pose-IO posed the participant's wrist at a target angle (Figure 83) and held it stable for one second, played a sound, and dismissed the pose, causing the hand to drop. Then, the participants' task was to recreate the previous pose. When satisfied, participants pressed a keyboard button, which caused the system to record the trial. For each trial, we recorded pose-IO's accelerometer reading during the target pose and during the recreated pose.

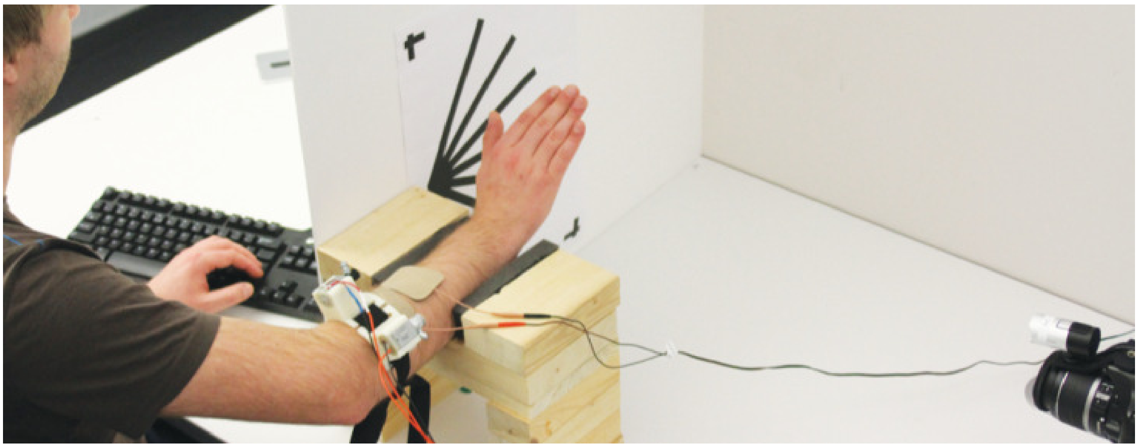

Figure 83: Apparatus: a barrier prevented participants from seeing their wrist.

4.1.5.2 Apparatus

As shown in Figure 83, participants wore pose-IO's electrodes on their dominant arm. The device was connected via USB to a notebook computer that ran the study script.

We recorded all trials using a camera. An opaque screen prevented participants from seeing their hands. To make sure that participants would complete the task based on their proprioceptive sense alone, we also masked out the "tingling" sensation caused by EMS by (1) lowering skin resistance using conductive gel and (2) attaching two vibration motors to the electrodes to create a sensation similar to EMS feedback [122].

4.1.5.3 Procedure

There were 7 target angles in 20° steps between -60° (flexion) and +60° (extension) reflecting the biomechanical constrains of the human wrist [25]. Each participant performed (7 target positions × 4 repetitions) = 28 trials. Target angles were presented in random order. Prior to the first trial, we calibrated the device for the respective participant. However, participants received *no* training on the task, as our expectation was that the unified correspondence between input and output would allow participants to complete the task based on their natural sense of proprioception alone. The study took about 30 minutes per participant.

4.1.5.4 Participants

We recruited 10 right-handed participants (3 female) from our local organization.

4.1.5.5 Results

Participants recreated poses with an average error of 5.8° (*SD* = 5.1°) across all trials.

A linear regression found the overall model fit to be R2 = 0.954. As shown in Figure 84, tilting the hand up further (shown on the left of Figure 84) resulted in slightly larger errors (*mean* = 6.49, *SD* = 5.93) (*mean* = 5.19, *SD* = 4.18). Furthermore, we found no statistical significant difference between any of the angles. For simplicity we present the results for the pairs with larger deviation, e.g., +40° vs. -60° (Z=-1.886; p=0.059) and +60° vs. -60° (Z=-1.274; p=0.203).

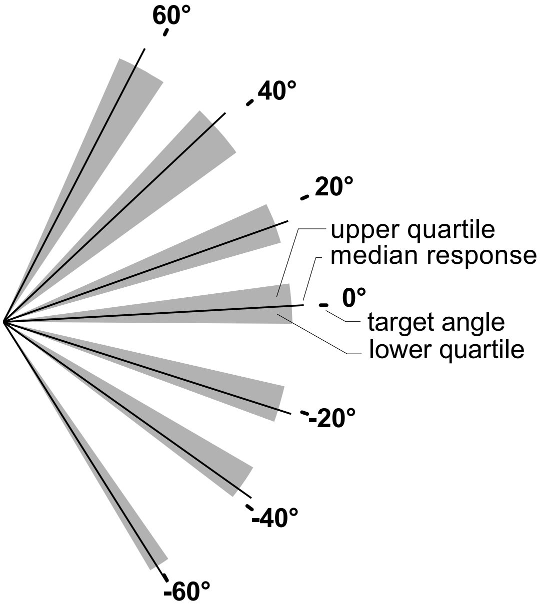

Figure 84: Differences in angle between target pose and recreated pose.

Interestingly, we observed several participants recreating not only the target wrist angle, but also the entire hand pose including the artifacts caused by electrical stimulation, such as the slightly overextended middle finger (Figure 85). This suggests that proprioception was at work on a broader scale.

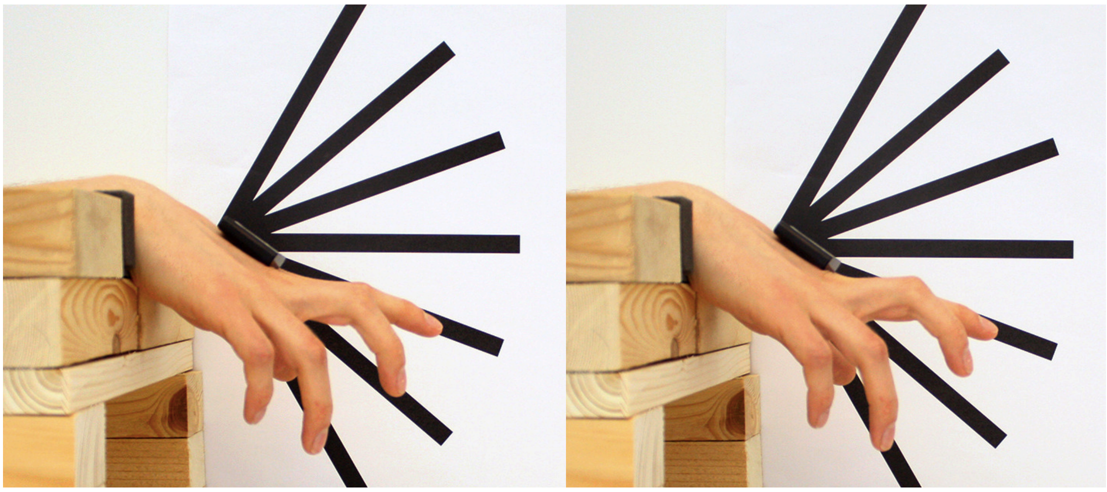

Figure 85: (a) Wrist actuated by pose-IO. (b) Pose recreated by participant.

4.1.5.6 Discussion

Our results show that the interaction did actually use and benefit from proprioception. As observed participants re-posing was accurate and included reposing of fingers. The results suggest that proprioceptive interaction's ability to allow for symmetric input & output offers benefits when it comes to recreating poses, as users can perceive and re-pose their wrist using their proprioceptive sense.

On a practical level, participants acquired the target poses with comparably high accuracy, i.e., 5.8° error on average. With respect to the motion range of -60° to +60° = 120° this represents an error of under 5%. If we map these values back to the motivating example of video scrubbing, this means that pose-IO would allow users to jump to a spot in a 1-minute video clip with ±3 seconds accuracy, which we argue to be a useful level of performance. Obviously, these values were obtained under idealized conditions, i.e., user was sitting down and immediately recreated the pose, and thus it should be interpreted as a lower bound for error.

4.1.6 An exploratory study of the proprioceptive interfaces

While our discussion so far was focused on the technical aspects and abilities of proprioceptive interaction, one of its aspects that we are personally intrigued about is its way of representing the computer as part of the user's body. Very unlike traditional human computer interaction systems, where computers are outside and different from the user, proprioceptive interfaces make the user's limbs *themselves* serve as the interface, which we argue is the very essence of proprioception—after all, the word stems from Latin "proprius", meaning "one's own". Our video-scrubber application makes the hand partially "owned" by the user and partially owned by the machine. The two games we showed earlier push this approach even further by questioning that notion of "ownership" in that they cause one's hand to be fully owned by the machine.

We were wondering how users perceive this aspect of proprioceptive interaction and what emotional response it might produce. This is what we investigated in this second, exploratory study. We had participants play the red hands game described earlier. We recorded participants' response to the interface on video and had them fill in a questionnaire.

4.1.6.1 Setup

We recruited 12 participants (3 female) from our local organization, which did not partake in the previous study, and asked them to play the red hand game for about 5 minutes. As discussed earlier, participants' objective was to evade getting slapped by the computer-controlled hands. All participants received candy as a small incentive to participate and we promised additional candy to whoever would score highest.

The game offered levels of increasing difficulty. As mentioned earlier, these were implemented in the form of barely noticeable actuation of the slapping hand before the actual slap (750 ms – 50 ms, depending on level). We did not tell participants about this "warning" mechanism, making it part of the game to either consciously or unconsciously figure it out. For this study we used an earlier pose-IO prototype, i.e., same hardware but not wearable yet.

To make sure participants only responded to the pose interaction, we canceled out any auditory signals by making them wear noise-cancelling headphones that played music. Each study session started by participants calibrating pose-IO, familiarizing themselves with the game mechanics, and playing one training level with 10 slaps. They then played 5 levels with about 10 slaps each. If they got hit less than 3 times per level, they proceeded to the next level and their score continued to go up. If not, their score stopped increasing, but we still let them finish the remaining levels to give them a chance to experience the complete game.

4.1.6.2 Findings

We had designed the game to be challenging and participants took the game quite seriously. All except one participant completed level 3; three level 4, and two made it to the fifth and final level. Participants rated the first two levels as easy (*mean* = 2.0 of 5; 1 = very easy, 5 = very hard) and the last levels as hard (*mean* = 3.92 of 5).

All participants rated the game as fun (*mean* = 4.6 of 5) as depicted in Figure 86 and further illustrated by Figure 87.

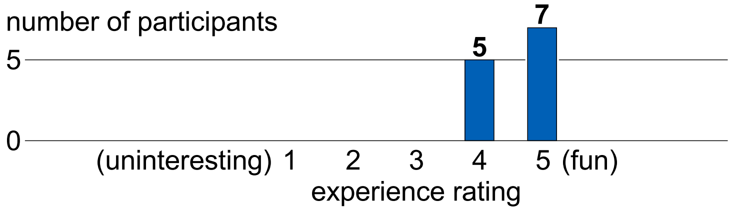

Figure 86: All participants rated EMS red hands as fun (1: uninteresting, 5: fun).

When asked why, seven participants pointed to the fact that they felt as if they were "playing against themselves". Several participants went back and forth between referring to the hand wearing the device as "me" and as "computer", such as "it is weird that you lose against yourself" and "sometimes the computer hand was faster than I".

In the last two levels, ten out of twelve participants stopped looking at their hands while awaiting the slaps. When asked about it, only two stated to have played by always looking at their hands, while five participants stated to play eyes-free and seven stated to have rarely looked. This suggests that participants operated our application by means of proprioception alone.

Most participants agreed that the muscle output had contributed to their excitement (*mean* = 4.91 of 5). One of the two participants who had continued to look at their hands explained: “it was so remarkable to see my hand moving without my intention that I could not look away”.

In summary, watching the twelve participants (Figure 87) confirmed that pose-IO affords being operated eyes-free and that users perceived the “blurred ownership” of the user’s body as compelling.

Figure 87: Participants playing EMS red hands (image with participants’ consent).

4.1.7 Conclusions on proprioceptive interaction

In this investigation, we proposed the concept of proprioceptive interaction, which we instantiated through a wearable prototype. Proprioceptive interaction leverages the user’s proprioceptive sense for both input & output, i.e., the use of pose as a bidirectional communication channel between human and computer.

We demonstrated how proprioceptive interactions allows for (1) eyes-free and ears-free use; (2) invoking the device at any time and returning to their primary task immediately when necessary; and, provides (3) two modalities: symmetric, in which input and output occur in the same limb, and asymmetric (i.e., one limb for input, another for output).

A unique aspect of proprioceptive interaction is that it represents the computer using a part of the user's body, which our second study suggests that participants found this experience compelling. On a philosophical layer, this "blurred ownership" manifests itself because operating a device only through proprioception thins the boundary between human and computer, since the machine partially inhabits the human. In the case of the presented proprioceptive interface, the muscles which control the wrist are the interactive device, in contrast to most input & output devices which are external pieces of hardware, sitting outside the user's body.

This investigation was, however, centered only on the relationship between the user and their own EMS-animated limbs. On our next project, we explored whether this EMS-based animation could breath a new life into *everyday objects* that the user is actively manipulating.

4.2 Affordance++

Affordance is a key concept in usability. When well-designed objects "suggest how to be used" [42], they avoid the necessity for training and enable walk-up use. Physical objects, for example, use their visual and tactile cues to suggest the possible range of usages to the user [42]. Unfortunately, physical objects are limited in that they cannot easily communicate use that involves (1) motion, (2) multi-step processes, and (3) behaviors that change over time. A spray can, for example, is subject to all three limitations: (1) it needs to be shaken before use, but cannot communicate the shaking, (2) it cannot communicate that the shaking has to happen before spraying, and (3) once the spray can is empty, it has no way of showing that it cannot be used for spraying anymore (and instead should now be thrown away).

As pointed out by Djajadiningrat et al., the underlying limitation of this type of physical object is that they cannot depict *time* [30]. The spray can is inanimate. Motion, multi-step processes, and behaviors that change over time, however, are phenomena in time. One way of addressing the issue is to provide objects with the ability to display instructions, e.g., using a spatial augmented reality display [126]. To offer a more "direct" way for objects to communicate their use, researchers have embedded sensors and actuators into objects, allowing them to be animated [83, 151]. This approach works, unfortunately, at the expense of substantial per-object implementation effort.

In this investigation, we propose a different perspective. While animating objects allows implementing object behavior, we argue that affordance is about implementing *user behavior*. The reason is that some of the qualities of an object are not in how the object behaves, but in how the user behaves when making contact with the object. A good part of the process of communicating how the user is supposed to operate the object, however, takes place before users even touch the object. Users operating a door handle do not just touch the handle to then re-adjust their hand position based on the handle's tactile properties; rather, the object communicates its use while the user's hand is approaching it. The haptic quality of the handle itself ultimately does play a role, but by the time the hand reaches the door handle, the user's hand is already posed correctly to grip the handle.

We therefore propose creating object behavior by controlling *user* behavior. We achieve this by instrumenting the user, rather than the object. This allows us to implement object behavior not by providing objects with the ability to respond to the user, but by re-creating how the object wants the user to respond to the object.

4.2.1 Affordance++: allowing objects to communicate their use

We call this concept of creating object behavior by controlling *user* behavior *affordance++*. Conceptually there are many ways of implementing affordance++, generally by applying sensors and actuators to the user's body, such as the arm. In this project, we actuate users by controlling their arm poses using electrical muscle stimulation, i.e., users wear a device on their arm that talks to the user's muscles by means of electrodes attached to the user's arm (we describe the device in detail in section "Prototype"). This allows for a particularly compact form factor and is arguably even more "direct" than the indirection through a mechanical system. However, the concept of affordance++ needs not to be tied to a particular means of actuating the user, but to the concept of doing so instead of actuating the objects that the user interacts with.

Figure 88 illustrates affordance++ at the example of the aforementioned spray can. Affordance++ allows the spray can to produce a range of different types of behavior. In the shown example, when the user grasps the spray can, the spray can causes the user to shake it. Our prototype implements this either using an optical tracking system or using a sensor in the user-worn device that recognizes a marker inside the spray can. Once recognized, the prototype plays back the desired behavior into the user's muscles.

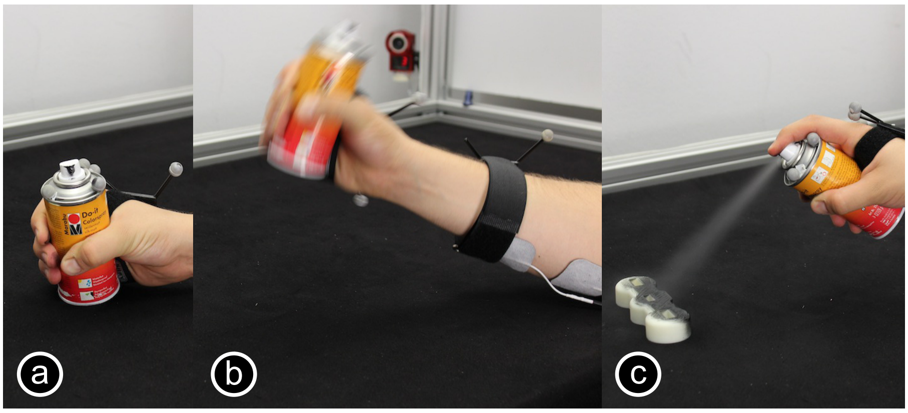

Figure 88: (a) This spray can needs to be shaken before use. (b) Affordance++ allows the spray can to make the user shake it before use. It implements this by electrically stimulating the user's muscles. (c) The can is "willing" to be used.

As we illustrate in the following section, affordance++ allows objects to produce multiple types of behaviors, including behaviors that start prior to physical contact. By storing information about objects' states, affordance++ allows implementing not only motion, but also multi-step processes and behaviors that change over time. On a technological level, the electrical muscle stimulation technology we use is able to *make* users perform certain motions. However, affordance++ intentionally avoids this and instead *suggests* how to use objects by actuating the user's hand with low intensity. This keeps users in the loop, allowing them to decide when to follow a suggestion and when to overwrite it.

4.2.2 Illustrating some use cases of affordance++

We now demonstrate the expressiveness of affordance++ at the example of six unfamiliar household objects and tools. All six of them we examine further in our user study. We cluster examples by the key features of affordance++.

(1) Motion: where to push the lamp? In Figure 89, affordance++ helps users operate a lamp of a rather abstract design [9]. (a) Here the user's hand approaches the lamp from the left side. (b) The object responds by gently pushing the user's hand towards the right, (c) where it flexes the hand gently to suggest pushing down. This tips the lamp over, (d) turning it on. In the same manner, when approaching the lamp while it is on, affordance++ moves the user's hand to the side and pushes it down to turn the lamp off.

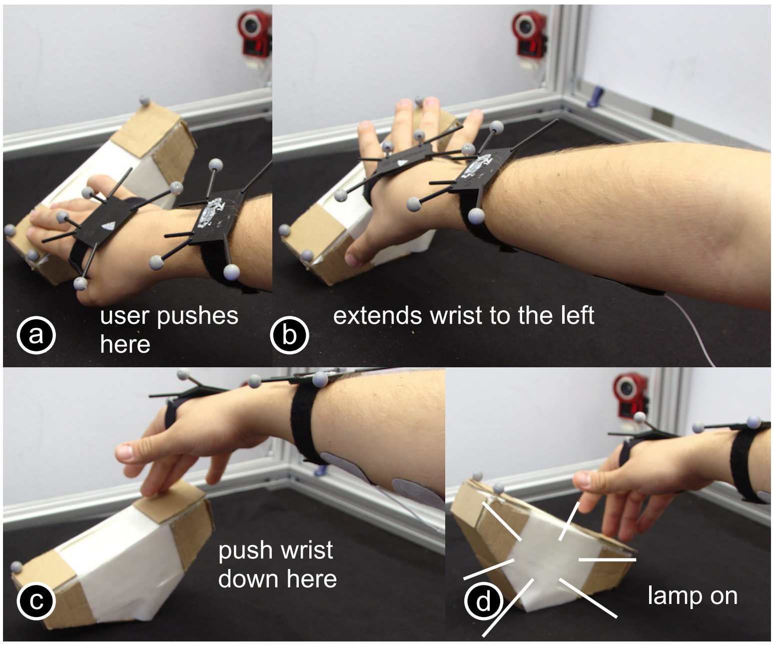

Figure 89: Here, affordance++ helps the user interact with an unfamiliar lamp.

(2) Multi-step processes: Magnetic Sweeper & Slicer. In Figure 90, affordance++ helps users handle an unfamiliar object. This "nail sweeper" allows users to pick up and drop objects with the help of a magnet—an example of a multi-step process. (a) The tool suggests grasping the handle by repelling the user's hand when trying to grasp any other part. (b) Afterwards the device suggests sweeping the nails by slowly rocking the wrist back and forth.

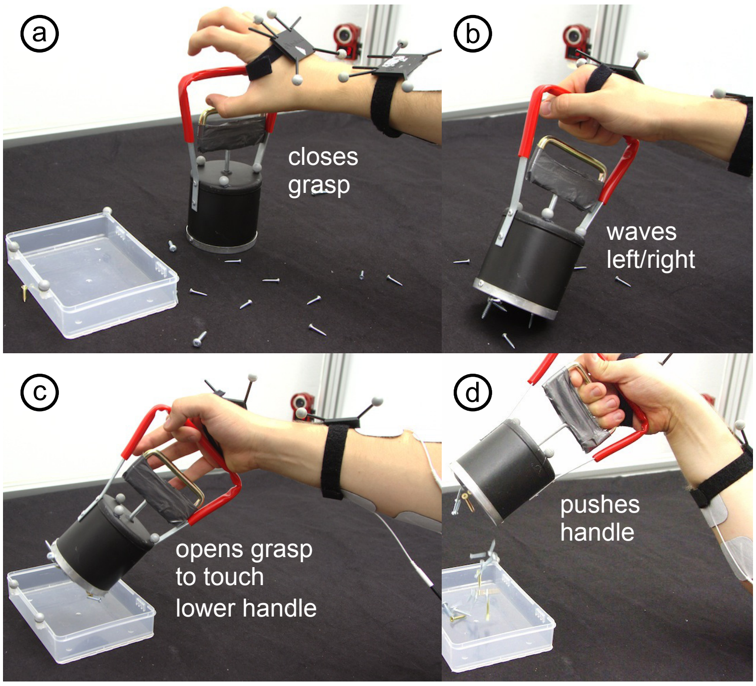

Figure 90: Example of a tool that requires a correct order of execution.

The critical moment occurs when collecting a screw. Here users typically reach for the lever below the handle, as they assume this is how one *collects* the screws. This assumption is, however, false and affordance++ repels the user's hand from grasping this handle and continues the sweeping motion. (c) Only when the user hovers over the container, affordance++ loosens the user's closed fist, allowing the user to grasp the lever and (d) by pulling the lever, the magnet releases the screws.

The next example of the multi-steps category, depicted in Figure 91, is a patented kitchen tool with multiple functionalities. The challenge here is to find out what each part does and when to use it. (a) The user explores the unfamiliar tool and tries to grasp it. Here, affordance++ repels from grasping the knife blade and only affords grasping the other end. (b) After grasping, affordance++ suggests cutting with the blade, by gently rocking the wrist back and forth.

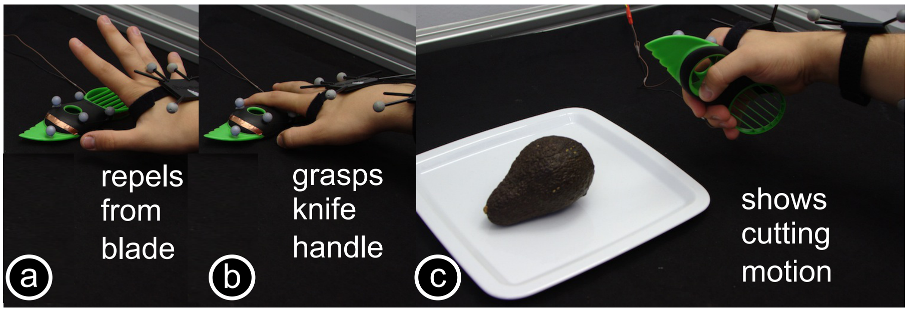

Figure 91: Cutting the avocado in half with this tool.

The next step is to remove the pit from the avocado, shown in Figure 92. This kitchen tool affords this by providing a set of blades inside a hole that extract the pit. (a) Affordance++ repels the user from removing the pit with the knife tip and (b) waves the tool back and forth parallel to the pit. This suggests a slamming motion, which the user performs in order to extract the pit.

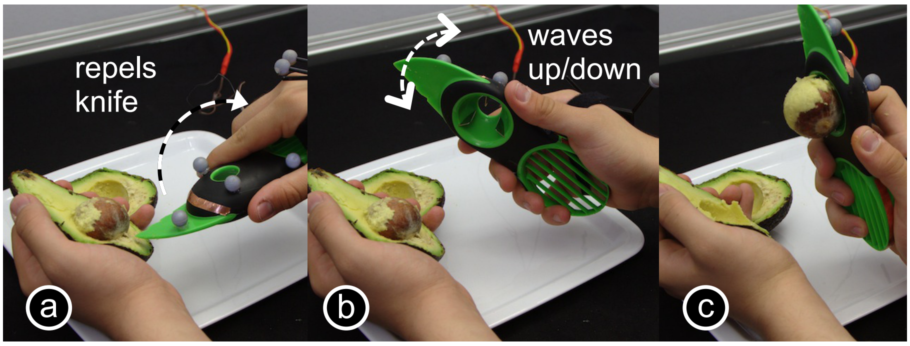

Figure 92: Extracting the pit using the tool.

The last step is to slice the avocado in pieces. While the conventional way is to peel off the skin and then slice each piece individually with a knife, this tool does it in one step. However, as we found in our user study, this instruction is not easily discoverable. Affordance++ helps here by (a) releasing the grasp gently as to suggest grasping the other end and (b) moves the tool towards the avocado suggesting a scooping motion. (c) Users respond by performing the scoop, which slices the avocado simultaneously.

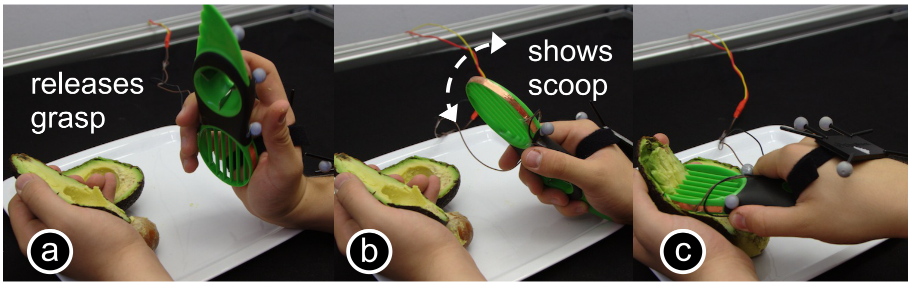

Figure 93: This tool allows for slicing the avocado without peeling the skin.

(3) Behaviors that change over time, e.g., the hot cup. The same mechanism applies to familiar objects for which the current state is unknown because object properties change over time. Figure 94 depicts how affordance++ supports a change in an object's property—here, temperature. (a) The user is trying to grasp the cup around its body while pouring hot tea. Albeit the object's typical affordance (i.e., round and small—affording grasping) this is a case in which the affordance needs re-adjustment. (b) Affordance++ now repels the user by over-extending the palm as to prevent grasping around the body, and (c) affords grasping only around the handle, by making the user thumb flex inwards into a pinch gesture.

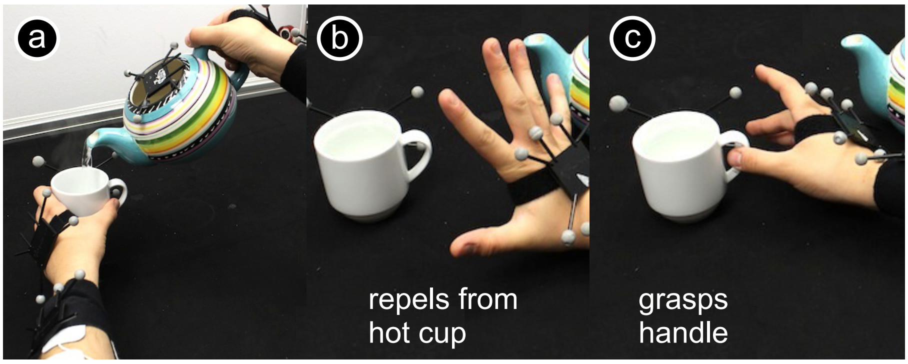

Figure 94: Changing behavior according to the object's state: (a) this cold cup affords grasping, (b) when hot it repels the user's hand, but (c) its handle continues to afford grasping.

Figure 95 shows a door that changes its behavior in accordance to what happens behind the door. The user approaches a door handle, ready to open it. (a) The room is busy however, as a meeting is taking place inside. The handle therefore repels the user's hand preventing it from grasping the handle and instead (b) suggests knocking by closing the user's fist and (c) rocking the wrist back and forth. The user (d) decides to knock by approaching the door's surface with the hand.

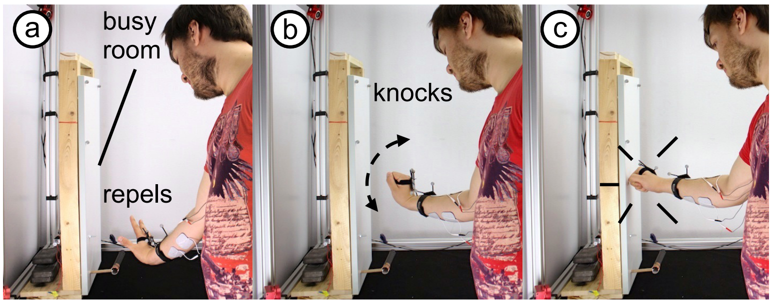

Figure 95: A door that repels using affordance++.

This "knocking example" is what we call a *preemptive demonstration* of how to operate the object. Figure 96 extends this concept further. Here, the door communicates with the user at different distances. At 75 cm, users feel their hand closing into a fist. (b) At 50 cm users feel their hand starting a gentle knocking pattern. Lastly, (d) at about 25 cm the knocking pattern becomes stronger. This is an example of attaching an envelope to an object. Here, the envelope maps to the amplitude of the knocking gesture. Another design is to affect frequency, i.e., the closer the user gets, the faster the knocking becomes.

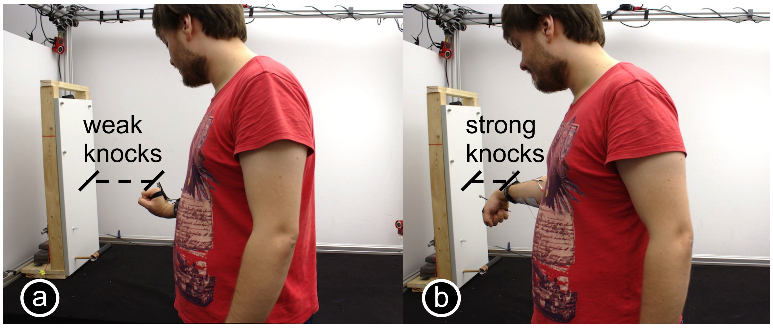

Figure 96: While approaching a door with the intention to knock, affordance++ allows the gesture amplitude to be communicated as a function of distance from user to object.

In Figure 97, the user is trying to open a door. When grasping the handle, affordance++ suggests turning left and prevents the user from turning the wrong way. Affordance++ is helpful here, because the mechanism is not visible to the user, which is the case for most push/pull doors [114].

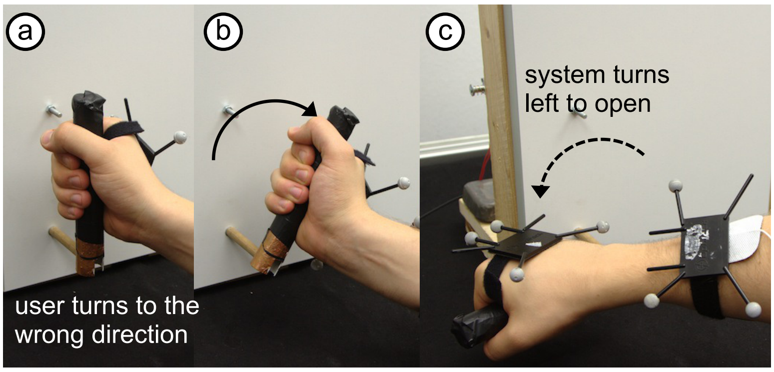

Figure 97: Overwriting the user's intention by providing the right turning direction.

Likewise, this example can be done at a distance with preemptive demonstration. While approaching a door's handle, affordance++ demonstrates the turning left/right (or pulling/pushing) prior to contact with the object.

4.2.3 Contributions, benefits, and limitations

Our main contribution is that we extend the notion of affordance so as to include dynamic object behavior. We thereby allow objects to communicate motion, multi-step processes, and dynamically changing behavior.

Affordance++ allows objects to *communicate* dynamic use to users, but at the same time keeps users in the loop, allowing them to overwrite objects' suggestions. A key property of affordance++ is that it allows objects to affect users *before* they make physical contact with the object. We demonstrate a simple prototype that accomplishes this by actuating the user using electrical muscle stimulation. We validate our concept in a user study.

On the flipside, affordance++ is subject to several limitations. (1) Our muscle-based implementation is low bandwidth limiting us to simple poses and simple behaviors. Also, any solution based on electrical muscle stimulation requires recalibration once the muscles fatigue over time [73]. (2) Our implementations are just simple prototypes to illustrate the concept. Future versions should integrate more elaborate tracking solutions, e.g., based on mobile optical tracking, such as the aforementioned Digits device [81].

4.2.4 Prototypes

Figure 98 shows the simple prototype setup we used to explore the concept of affordance++. It uses electrical muscle stimulation, which provides us with a particularly direct way of instrumenting users. The tracking component of the shown version is based on optical motion capture.

Figure 98: This affordance++ prototype uses an *Optitrack* setup for tracking.

Our prototype implements object behaviors by actuating the users' muscles using a computer-controlled electrical stimulation unit. The unit is comprised of four individually addressable channels. The unit allows for safe operation by limiting the output current to 100 mA. The pulse-width (150µs-250 µs) and frequency (80-140 Hz) are variable and calibrated per-user, per-channel and remain fixed once calibrated. Intensity is regulated on the fly using an *Intersil X9C102* digital potentiometer that addresses the voltage-controlled amplifiers. A microcontroller (*ATMEGA328*) steps the digital potentiometer up/down on request by the affordance++ software. The latency of a one-step change is around 1 ms, which makes the latency of a minimum-to-maximum actuation sweep less than 100ms. Depending on the pose and on the user's skin resistance [19], the object uses a different waveform (frequency and amplitudes), which is sent to the microcontroller and which then regulates the output current.

4.2.4.1 Muscles actuated

Figure 99 depicts all the poses we use in affordance++ and the muscles actuated for each pose: *squeeze*, *turn*, *repel*, *drop*, *shake* and *raise*.

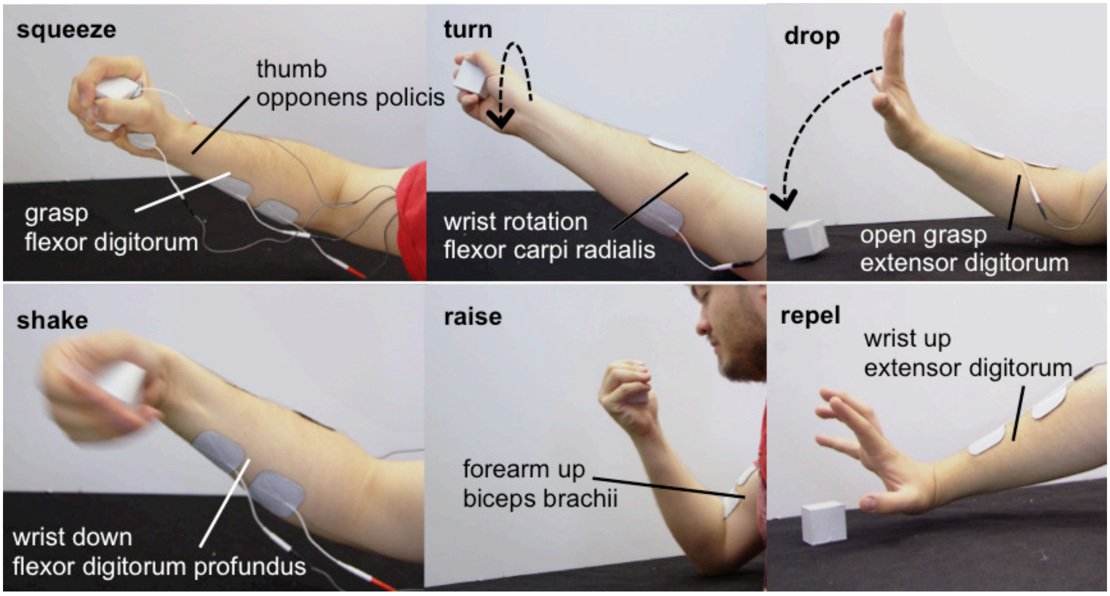

Figure 99: Electrode placement for each gesture used in affordance++.

Each action uses at most four electrodes. Complex actions are achieved by combining these actions, for instance, "knock on a door" is composed of *squeeze* for a closed fist and *shake* for a knocking motion.

4.2.4.2 Tracking and how it triggers object behaviour

The prototype triggers object behavior when it detects certain spatial relationships between the user's hand and one or more tracked objects. Our prototype computes this by encapsulating tracked objects and hands with collider objects (Figure 100). When the user's hand intersects with an object's collider, it triggers the respective object behavior. By using colliders of different sizes, behaviors may be triggered closer or further.

The prototype allows triggering object behaviors based on the hands proximity to the object, as well as when the user is reaching for the object. The prototype computes the latter by extrapolating the hand's trajectory using a ray it casts in the direction of the hand's motion. It determines the direction of motion based on the moving average of the past hand positions (from the last 50ms = 20 frames).

Furthermore, this prototype provides users with two strategies to dismiss an ongoing object behavior. First, by removing their hand from the collider volume. Secondly, if the ongoing behavior involves motion (such as knocking the door) or in the case of handheld objects (e.g., spray can) users may also dismiss it simply by resisting the motion. Our prototype implements these mechanisms and behaviors based on the Unity3D engine.

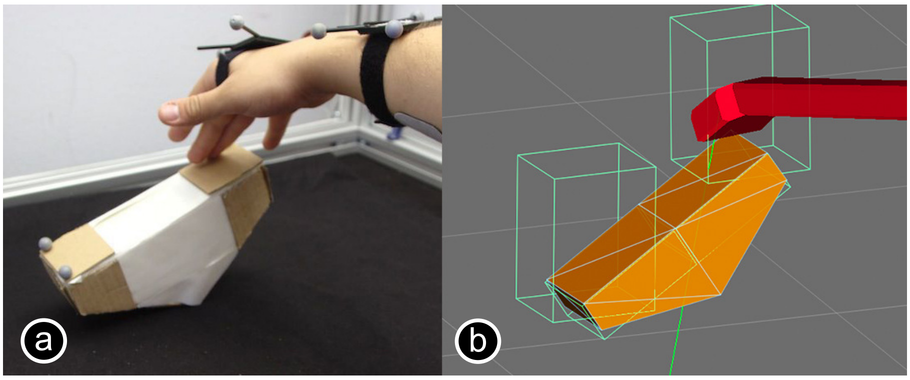

Figure 100: (a) Here the user approaches the designer lamp. (b) In Unity3D, we detect the collision with the bounding volumes and trigger the pulling down gesture.

This prototype works reliably, but it is obviously limited to the size of the shown workspace and subject to occlusion.

4.2.4.3 Mobile RFID-based tracking

To allow us to explore mobile use, we created a second prototype. Although limited to interactions that can be expressed with proximity using RFID, it allows us to prototype some of the examples presented earlier, such as the door in Figure 95, by placing an RFID tag on the handle.

As illustrated by Figure 101, the prototype contains all electronics attached to a sleeve. It features a 2-channel version of our aforementioned computer-controlled muscle stimulator, which it controls using digital potentiometers controlled by an *ATMEGA328* microcontroller, which sits behind the RFID antenna. The prototype contains the SM130 Mifare 10 MHz reader and objects contain passive tags.

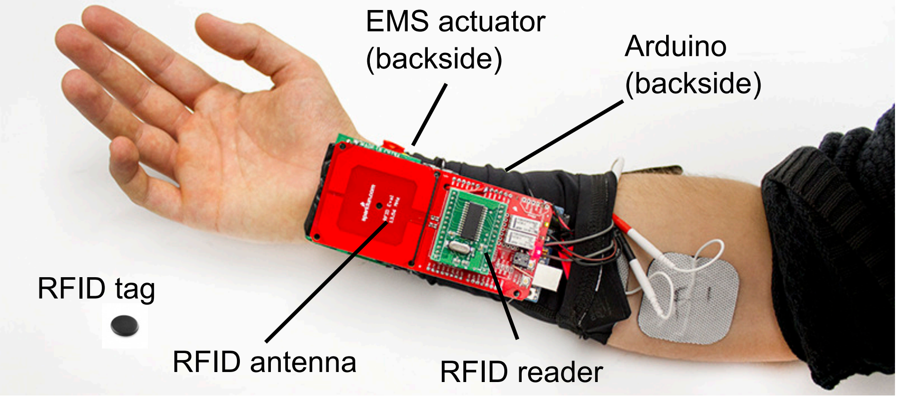

Figure 101: Wearable prototype based on RFID.

4.2.5 User study

In order to validate the concept of affordance++, we conducted a user study.

The objective of Task 1 was to verify that affordance++ indeed allows communicating identifiable object behaviors. For this purpose, we had participants touch blank generic objects that played back different types of behaviors. Participants responded by naming the behavior they felt the object was performing.

The objective of Task 2 was to verify that affordance++ indeed allows objects to communicate use. We had participants operate six objects that attempted to communicate their non-obvious use through affordance++.

In the interest of clarity, we present procedures and results grouped by task.

4.2.6 Study task 1: understanding the meaning of induced motion

Each participant received the same blank, generic white cube shown in Figure 99, which we provided with one of six affordance++ behaviors: *repel*, *drop*, *turn*, *raise*, *squeeze*, and *shake*; Figure 102 depicts an example of participants exploring the objects' behavior. Participants were not informed any of these behaviors beforehand. All participants experienced all six behaviors; the order of behaviors was counterbalanced across participants.

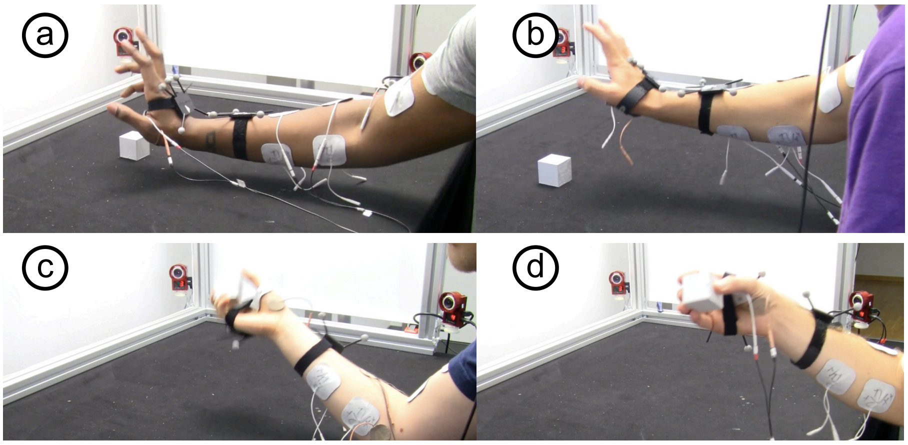

Figure 102: Four participants exploring (a,b) *repel* and (c,d) *raise*.

We gave participants two minutes to explore each behavior and encouraged them to think aloud. Participants then described "what they felt the cube wanted them to do" and rated how confident they were about this judgment on a 7-item Likert scale. We videotaped participant's responses.

4.2.6.1 Apparatus

We used the prototype described earlier to actuate users using EMS, but replaced optical tracking by a Wizard-of-Oz approach. Hence, the wizard visually confirmed the user's contact/approach with/to the object and triggered the predefined stimulation as defined in each trial (i.e., the behavior for that trial). This allowed us to avoid the use of markers, which we felt would have biased the study by suggesting grasping poses. It also guaranteed perfect "tracking".

Calibration of each pose took about 2 to 5 minutes and was conducted by the experimenter. There was no re-calibration during trials.

4.2.6.2 Experimental design

In order to prevent Task 2 from biasing the outcome of Task 1 (by providing additional context for the observed behaviors), we forewent counterbalancing and had all participants perform Task 1 first, then Task 2. The duration of each task was around 20 minutes (minus calibration time).

4.2.6.3 Participants

We recruited 12 participants (2 female) from our institution (*mean*=25 years old, *SD*=3.36). All were right handed. Participants received a compensation for their time.

4.2.6.4 Results

Overall, in 76% of all trials participants correctly named the behaviors the object had been designed to communicate. Figure 103 breaks down the results by individual behavior. In the remaining 24% of trials where participants did not name the exact behavior, they named some behaviors that were reasonably close, such as *juggle* instead of *shake*. Overall, this result suggests that affordance++ indeed allowed the blank cube to communicate identifiable object behaviors.

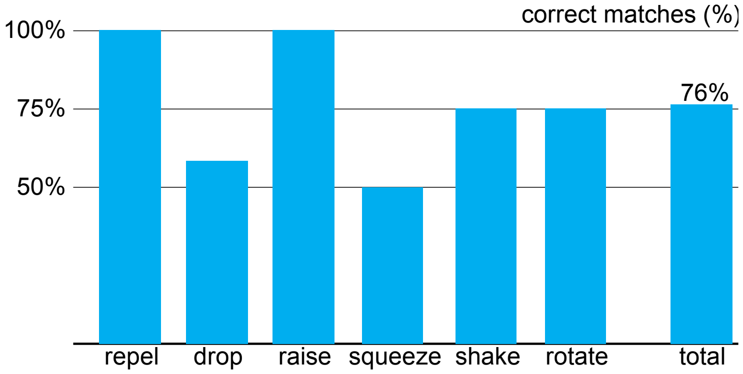

Figure 103: Correctly guessed behaviors, average for all users.

The most successful behaviors were *repel* and *raise* in that all participants correctly named them. These behaviors were also rated very high on confidence, as shown in Figure 104. Participants came up with the correct answer for *repel,* sometimes instantly when they first approached the object, as depicted in Figure 102. Six participants stated: “it does not want to be touched” one “I cannot touch it” and one “I’m not allowed to touch it”.

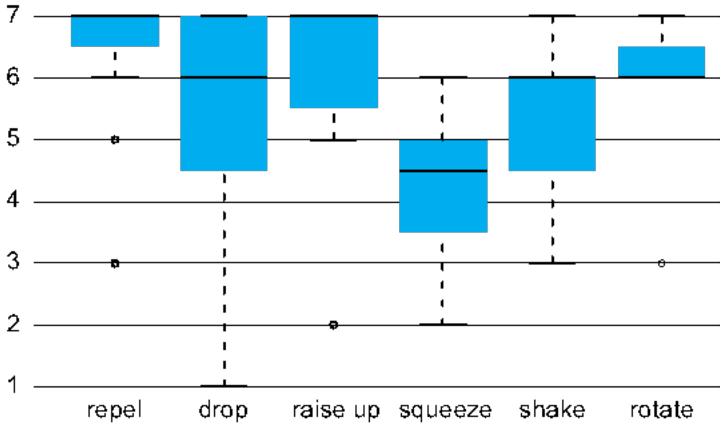

Figure 104: Confidence per behavior (Median and IQR, and total range of variation).

Raise was equally well understood by participants, however it often took additional exploration: "It wants to be lifted to my face, perhaps for eating" (6 participants), "throw it away" (P2), or "look closer" (5 participants). Two participants added that "raise" should elicit more muscles, such as the shoulder muscles for increased realism.

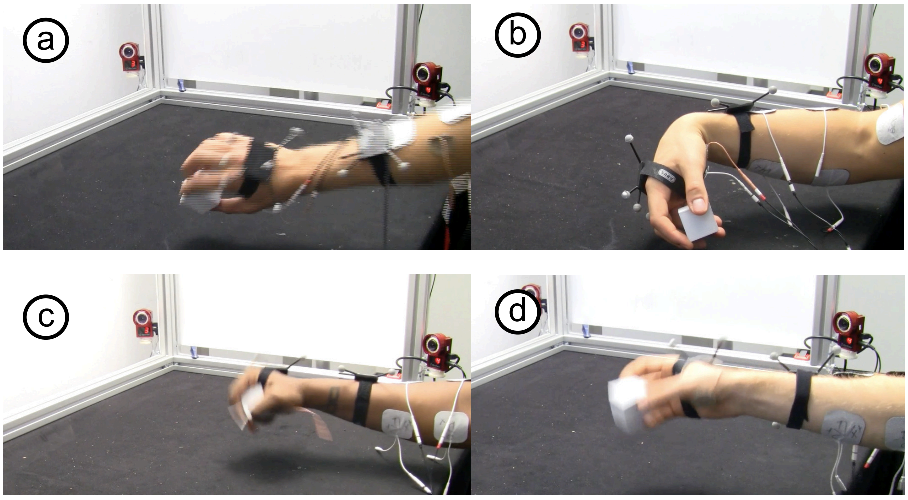

Figure 105: Four participants exploring (a,b) *shake* and (c,d) *rotate*. (b) This *Rotate* failed because the wrist flexed more than it turned.

Shake and *rotate* (Figure 105) were matched correctly in 75% of the trials, and were found to be understandable. Likewise, as for *repel*, we observed *shake* to be fairly quickly recognized, as most participants provided an answer within the few first seconds: "it moves me back and forth, like shaking" (P4) or "it's shaking" (P2).

Drop and *squeeze* (Figure 106) were found to be misleading, being guessed for 58% and 50% of the cases, respectively. Squeeze was rated less understandable than all other behaviors. The participants described most mismatches with regards to *squeeze* as: “a bit of turning” (P1), “a slight twist” (P3) or “rotating the cube in the palm using the fingers” (P9). As correct guesses we classified “to squeeze it” (P2, P4, P10), “grab harder” (P5), and “hold with more pressure” (P12). Analysis of our video recordings revealed that four participants’ *squeeze* pose was also characterized by some degree of wrist turning, caused by the palm flexor, which might explain the confusion.

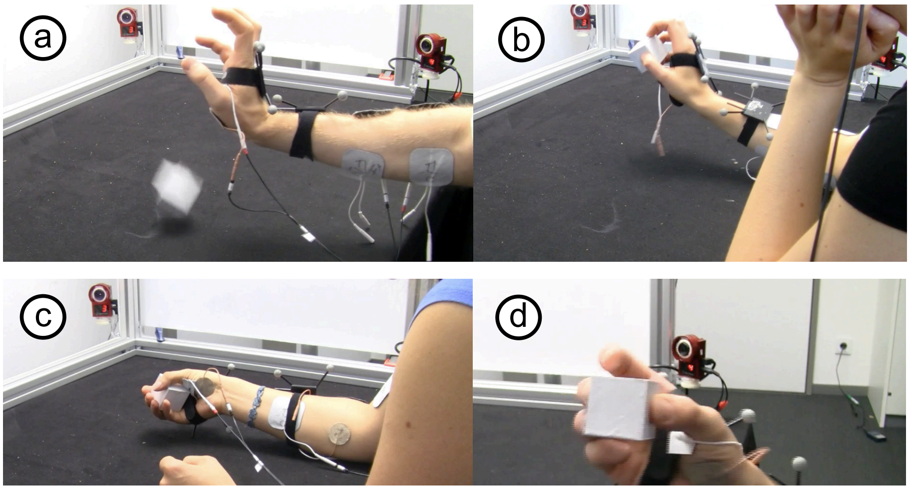

Figure 106: Four participants exploring (a, b) *drop* and (c,d) *squeeze*. (b) Here the muscle actuation was not sufficient to cause the participant to *drop* the object.

However, *drop* was still rated as understandable, because even if the cube did not properly fall from the closed grasp, the motion was found to still convey that the cube “did not wanted to be lifted” (P7) or “thrown like a dart” (P12). The correct matches we described as: “to let it fall” (P3, P2).

4.2.7 Study task 2: affordance++ communicating use

The goal of this task was to verify that affordance++ indeed allows objects to communicate use. We had participants operate six objects that tried to communicate their non-obvious use using affordance++. In particular, we wanted to assess whether affordance++ is able to communicate the three categories of behavior mentioned earlier, i.e., motion, multi-step processes, and dynamically changing behaviors.

Figure 107 shows the six objects used for each of the six trials. These are the six objects discussed earlier. They included three unfamiliar tools, i.e., avocado peeler with avocado, nail sweeper with nails, and designer lamp. All participants confirmed that they were unfamiliar with these objects. Next were three familiar objects that we had provided with new behavior, i.e., a door handle that opened to the left, a cup filled with hot liquid, and a spray can that required shaking before use.

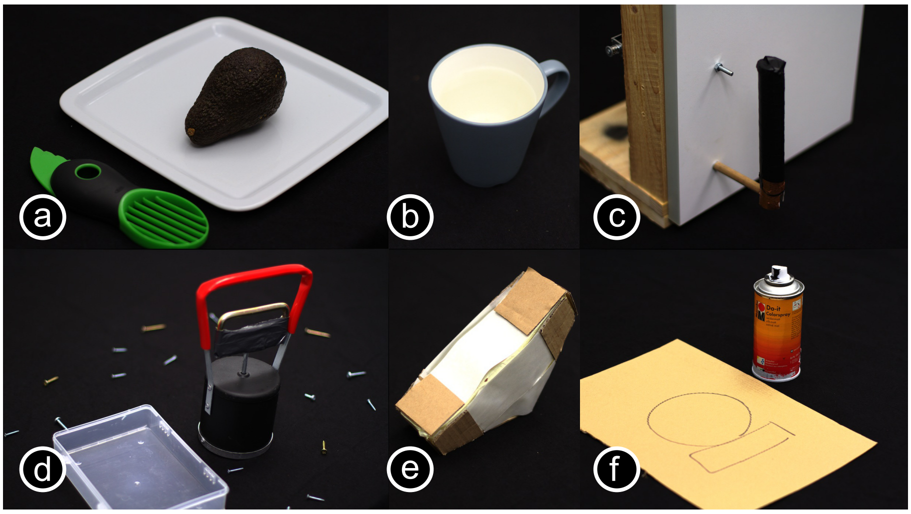

Figure 107: The objects used in Task 2.

4.2.7.1 Procedure

For each trial, we placed one of the six objects on the table in front of them (to prevent them from watching the setup, we blindfolded them until ready). We then asked participants to "prepare the avocado for eating in thin slices" (avocado tool), "place all nails into the container" (nail sweeper), "turn the light on" (lamp), "drink some water" (cup), "open the door" (door), and "paint this" (spray can). We encouraged participants to think aloud while interacting. Once the participant had completed the task, we asked: "please explain step by step what the object did to you." Note that the name of the object was never stated; we simply referred to it as "object". Participants also self-assessed how well they had understood the suggested use on a 7-item Likert scale, how much they felt that the object *itself* (i.e., without the actuation) had communicated its use (7-item Likert), and how much affordance++ had helped them discover the use of each tool for each trial (7-item Likert).

4.2.7.2 Results

Each of the twelve participants successfully figured out how to operate all six objects. We organize the results below accordingly to the three key features of affordance++: (1) motion, (2) multi-step processes, and (3) behaviors that change over time.

(1) Affordance++ communicates motion. All 12 participants, for example, shook the spray can. Ten of them stated that they had shaken the can because of the actuation and they would otherwise have forgotten to do so. One of the participants who stated to have not forgotten to shake added: "I found it useful that the system shakes the can for me, as a way of confirming what I was about to do, I might forget some other time, but this time I did not" (P4).

For the majority of participants that would have forgotten to shake, the first reactions were enthusiastic: "cool, I was about to forget" (P1). As shown in Figure 108, all participants agreed that the spray can itself had not (visually) conveyed the requirement to shake, but that they had understood the shaking motions. Lastly—probably in the light of their forgetfulness—participants rated affordance++ to be useful in this case.

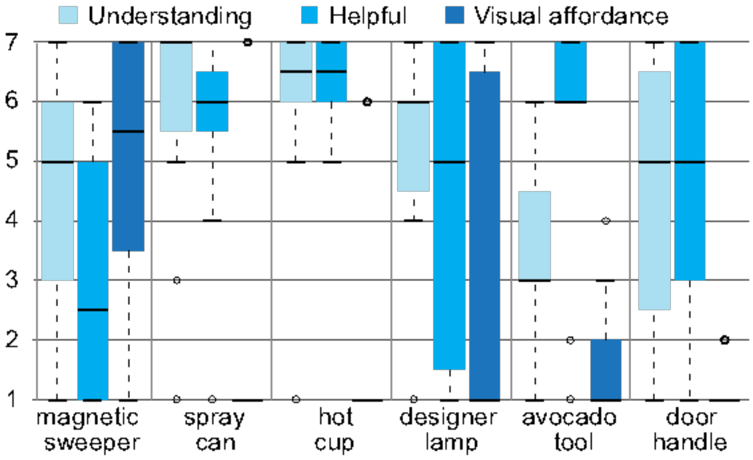

Figure 108: Participants' ratings for the six test objects.

Ten of the participants stated to have figured out how to turn the designer lamp on based on the induced motion. Comments included "this helps me very much, I had no idea" (P2), "(…) my intention would have been to push it like a button, but I couldn't find a button" (P9). The other two participants discovered the tilting mechanism accidentally by repositioning the lamp in all possible configurations or by mere chance. While most users rated the lamp's visual affordance as poor, two users rated it as 7 (i.e., easy to visually assess tilting) and added: "tilting was my first intuition, but by moving my muscles it confirmed it" (P3).

(2) Affordance++ communicates multi-step processes. All participants explored the avocado tool before touching the avocado. While reaching for the knife blade they got repelled, the actuation was suggesting them to grasp the other end. Then, affordance++ shook the wrist up and down to suggest a cutting motion, P3 stated: "I've seen this before, it means chop" and P7 said "I was confused, but then I got confident that it was a knife because of the gesture."

After participants had sliced the avocado in half, five of them released the knife by opening the grasp slightly and explored an alternative way to remove the pit. Five of them stated that the actuation had suggested doing so. These participants were fairly surprised when they realized that they needed to push the hole against the pit, which affordance++ had suggested by shaking the tool up and down: "when I put it on the pit it told me to push. It repelled [my hand] when I tried to use the blade" (P11), "I would have used the knife, but not the rest" (P6) and "It really helped me to use this tool because I had no idea [how to use it]" (P2).

All participants tried to peel the avocado using the knife, but were repelled by affordance++. After that, affordance++ opened the grasp to suggest that the tool should be manipulated in a different way. All participants responded accordingly and explored an alternative way to hold the tool, such as “normally I do it with a spoon, but it didn’t want me to cut it, which made me turn it and explore the tool, and then I saw [the blades] and felt how to use it” (P7), “The scooping tool told me to pull [points down towards the avocado]” (P11). Mainly due to the pit removal step, e.g., “all [the steps] worked out except this part in the middle, it really confused me” (P10), participants on average rated their own understanding of the actuation lower than for the other tasks (see Figure 108). Moreover, all participants agreed that their first visual exploration did not reveal how to use the tool.

The magnetic sweeper was the task in which participants rated the necessity for affordance++ lowest. This is likely explained by the fact that participants stated that the object *itself* already visually conveyed how to use it (see Figure 108), i.e., “looks like a magnet” (four participants). Furthermore, although they understood the stimuli during this task, they did not feel the need for affordance++ to collect all screws correctly.

Also, four participants found that the system corrected the order of the task: “first I wanted to touch the lower handle (…) I got repelled (…) and then it helped me (…) to grasp the upper one”. Furthermore, some participants assumed pulling the lower handle would collect the screws, which is incorrect because pulling the lower handle releases the magnet. When they did so, affordance++ repelled them from pulling it and they responded accordingly: “first thing [I did] was pulling it, the system pulled my hand [away] to prevent that. It was very clear and very helpful” (P8).

(3) Affordance++ provides dynamic affordance according to state. During the “drink some water” task, 11 out of 12 participants reached first for the cup’s body. They felt repelled from grasping the cup’s body and proceeded to the handle after some seconds. One participant added “it doesn‘t want me to not drink from it” (P10). Furthermore, another participant confirmed that the water was hot by attempting to dip the finger in the liquid, affordance++ repelled again and the participant stated “ah, I feel the heat now” (P7). The affordance++ poses used for this task, i.e., repel from body and grasp around handle, were found understandable by the participants (see Figure 108). All participants agreed that the cup did not provide a visual cue that it was hot. Participants were enthusiastic about how helpful affordance++ was for this situation and most added: “for hot liquids or any danger, is very helpful”.

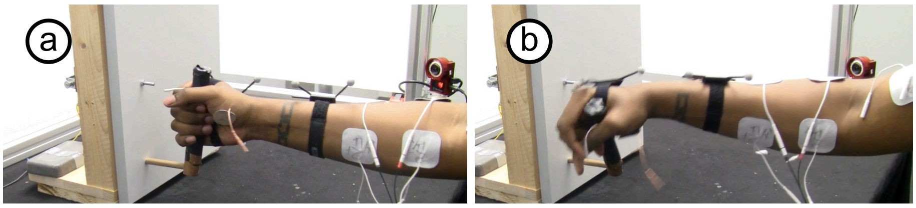

Figure 109: Example of overriding the conventional behavior.

The door handle worked in 9 out of 12 cases. Even with the non-standard turning direction (left instead of right) participants opened the door without forcing it right or pulling it: “I turned the lever to the left side. It’s not the normal direction” (P1), “It clearly gave me a direction to turn. (…) I would have tried around more stuff” (P4), and “I first tried in one way, and it pushed me to the other direction” (P12). Furthermore, all participants agreed that the door handle did not visually communicate to which side it would open. Lastly, participants found the system to be useful such as “if I would travel [to another culture] and find such a door, it would be helpful” (P12).

4.2.8 Conclusions on affordance++

In this investigation, we presented the concept of *affordance++*, i.e., an extension to the traditional notion of affordance that allows objects to also communicate (1) motion, (2) multi-step processes, and (3) behaviors that change over time.

Interestingly, because we actuated the user’s muscles instead of actuating the object, we expected users to perceive that they were the ones depicting the usage instructions. However, this is not what we observed in the user study. In fact, we observed that the agency (“who did what”) was redirected to the object—“*it* doesn‘t want me to not drink from it” (P10) is very different from “*I* should not drink from this cup”. The latter would elicit a more user-centered (or ego-centered, the “I”) understanding of affordance, rooted in "what can I do with this object". However, it seems that we observed, more often, the former. This projects the affordance to an object-centered perspective (the “it”). Furthermore, the aforementioned observations reported in Task 1 suggest that, to some extent, users “believe” that a script with an intention is attached to the object. For instance, six participants stated the cube “wants to be lifted to my face, perhaps for eating”. This ties together with the notion, developed by Latour, that humans communicate with the artifacts they operate [91]. Indeed, Verbeek states: “technological artifacts are able to exert influence on human actions, by means of non-lingual message” [153]. Comments such as “the tool *told* me to pull” (P11) suggest this is the case for affordance++, in which object-user dialogs do not happen on a verbal level (visual or auditory) but non-verbally, through the user's body motion.

It remains to be understood, for future work, whether another user actuation technique or even animating the objects themselves via mechanics, would shift this object-centered perspective to a different experience.

4.2.9 Reflecting on the quality of our information access via EMS

We pause the reader here for a moment to reflect on what we achieved so far in terms of interactive systems based on EMS for information access. Both pose-IO and affordance++suggest that users can access information by finding their body posed in a particular manner, both explicitly (pose-IO) and implicitly (affordance++).

Once we analyzed the resulting interactions that these prototypes enable, we realized these are fairly quick information accesses. For instance, while using pose-IO, a user might check the value of a particularly variable by invoking the device on their wrist; the user might then, decide to change the variable's value by also posing their wrist. This complete interaction is short and takes just a few seconds.

However, the information access tools we operate daily vary wildly in the information bandwidth let users access. On the low bandwidth side, tools such as a wristwatch allow us to access small pieces of contextual information in quick flashes of interaction (the so-called micro-interactions [7]); because their bandwidth is limited, the amount of intellect augmentation [33] their provide is also limited. On the other hand, we often employ higher bandwidth tools when we undertake sense-making activities, such as when we analyze data using digital spreadsheets.

Now looking back, with a critical eye, at the interactive systems based on EMS we have built and presented so far, we see that these cluster themselves closer to micro-interactions (quick & low bandwidth information access) than to sense-making activities.

Thus, we felt this called for a targeted investigation on precisely this question: what is missing to build a interactive system based on EMS that supports longer and more meaningful interactions that could even support sense-making activities?

4.3 Muscle-Plotter

The interactive EMS systems presented so far lack expressiveness. Our interactive EMS systems output a single 1D output variable, such as screen tilt (muscle-propelled force feedback) or wrist tilt (pose-IO), or one of n behaviors (affordance++). Since subsequent output overwrites earlier output, users never see more than a single value.

We now explore how to create more expressive EMS-based systems. *Muscle-plotter* achieves this by *persisting* EMS output, allowing the system to build up a larger whole. More specifically, as depicted in Figure 110, (1) muscle-plotter spreads out the 1D signal produced by EMS over a 2D surface by steering the user's wrist, while the user drags their hand across the surface. Rather than repeatedly updating a single value, this renders *many values* into curves. (2) By adding the pen, we persist this signal, allowing the system to build up a larger display, which in turn enables longer and more meaningful interactions. Muscle-plotter is a closed-loop EMS system. It allows users to enter information into a computer system by writing using an Anoto pen and it allows the computer system to respond by making the user plot. Muscle-plotter accomplishes this by actuating the user’s hand that is holding the pen by means of a medical-grade computer-controllable electrical muscle stimulator.

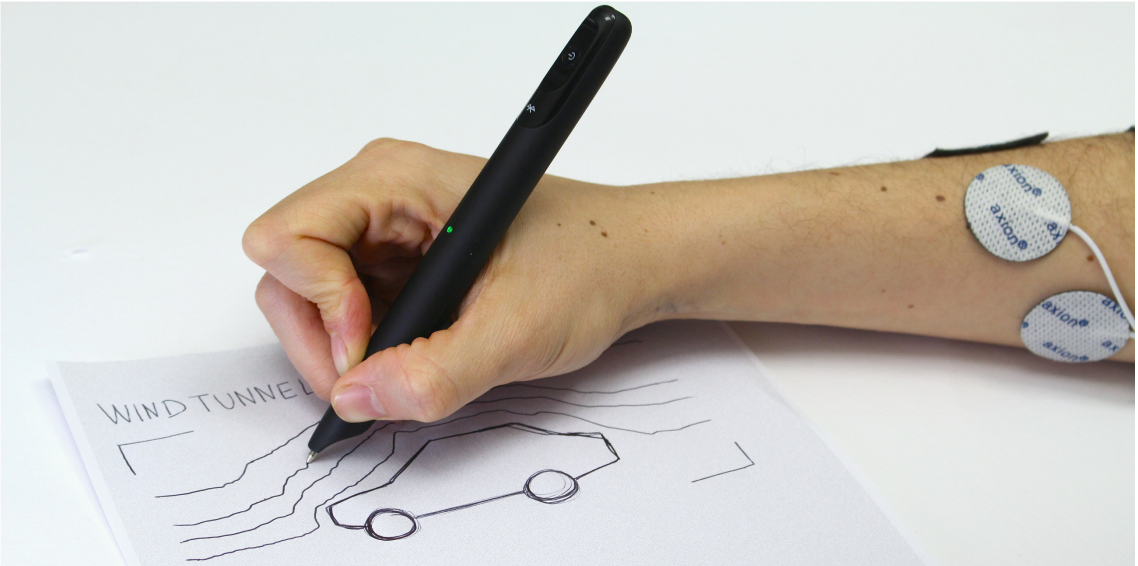

Figure 110: An interactive wind tunnel simulation with pen input and output, which is based on EMS.

4.3.1 Walkthrough: an interactive wind tunnel simulator

We think of muscle-plotter as a tool for mobile sensemaking in that it allows users to interact (input *and output*) with an intelligent backend.

Figure 110 shows an example of such a use case. Here, a car designer is iterating on the body of a new car, sketching it and analyzing implications of its design on the car's aerodynamics. The designer wrote "windtunnel" onto the paper and has drawn crop marks around the car. Since the user does so using a pen that offers built-in tracking (*Anoto* [150]), muscle-plotter "sees" this input. It recognizes the handwriting using a handwriting recognizer (*Tesseract* [115]) and forwards its output to a wind tunnel simulator running in our custom backend. The system computes the wind velocity field and makes it available to the pen frontend.

To plot the streamlines: this designer moves the hand to the left of the car sketch and sets down the pen. As the pen enters the wind tunnel simulator's bounding box, muscle-plotter starts sending electrical impulses to the user's wrist, which from now on continuously actuate the user's wrist. While the user moves the hand horizontally across the paper, muscle-plotter controls the hand's vertical position using a closed-loop control, resulting in plotted streamlines. Repeating this process produces a field of streamlines, allowing the user to judge the aerodynamic behavior of the current car design. Continuing the example from Figure 110, the designer is now contemplating whether the car should have a shorter rear and a rear door, also known as a *hatchback*. In Figure 111, the designer sketches one possible hatchback design and by having muscle-plotter draw streamlines on this new design, investigates what implications the change in body shape may have on the car's on aerodynamics.

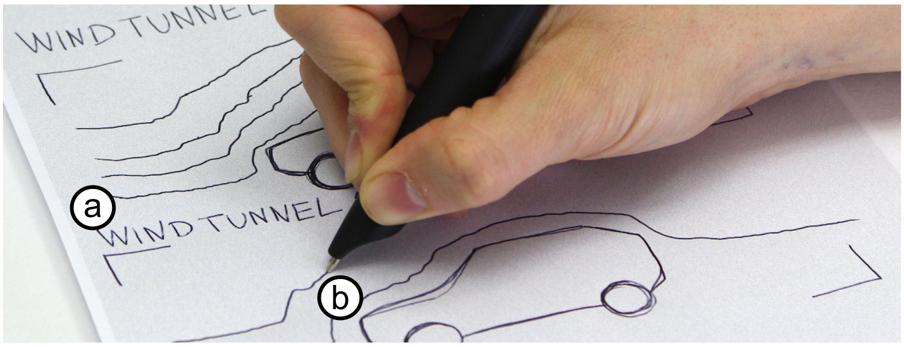

Figure 111: Selecting which sketches to simulate in the wind tunnel.

Line chart: Surprisingly, the hatchback's streamlines look straighter that the sedan's (Figure 112a), possibly suggesting an aerodynamic advantage. The user decides to drill down by plotting aerodynamic profiles of the tail winds of the two car designs. As shown in Figure 112a, the user draws one vertical line across each of the two cars' tail sections, and annotates them with "*crosssection sedan*" and "*crosssection hatchback*", thereby creating a series of data points for each car. As shown in Figure 112b, the designer now sketches a blank coordinate system, writes *"plot sedan*", sets down the pen down left of the coordinate system, and drags the pen into it. As shown in Figure 112c, muscle-plotter responds by plotting wind speeds across the cross section into the coordinate system. For comparison, the user now writes "*plot hatchback*" and plots the wind speed function for the hatchback into the same coordinate system.

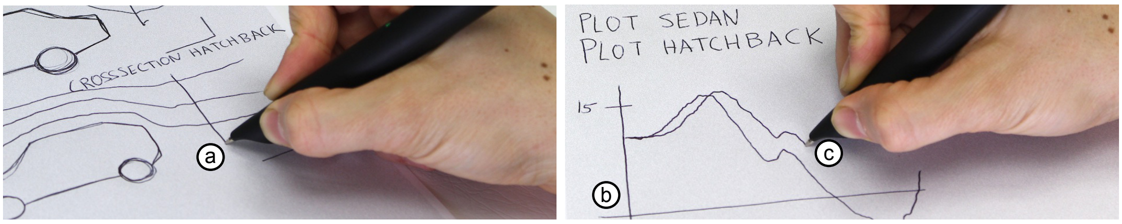

Figure 112: (a) The user defines a cross section of the streamlines. (b) Then, draws two axes of an X-Y plot, and (c) drags along the X-axis while muscle-plotter actuates the wrist to plot the Y values.

Zooming: As shown in Figure 113a, the right halves of the two plots look different. Values below the X-axis indicate negative wind speeds, which suggests undesirable turbulences. Surprisingly, again the hatchback seems to be performing better. The user drills down further by writing "*zoom tail*", drawing two crop marks that select the portion of the chart to re-plot, draws a fresh coordinate system, writes "*plot tail sedan*", and plots the close-up of the chart. The close-up clearly indicates that the rear profile of the hatchback is indeed subject to less turbulence.

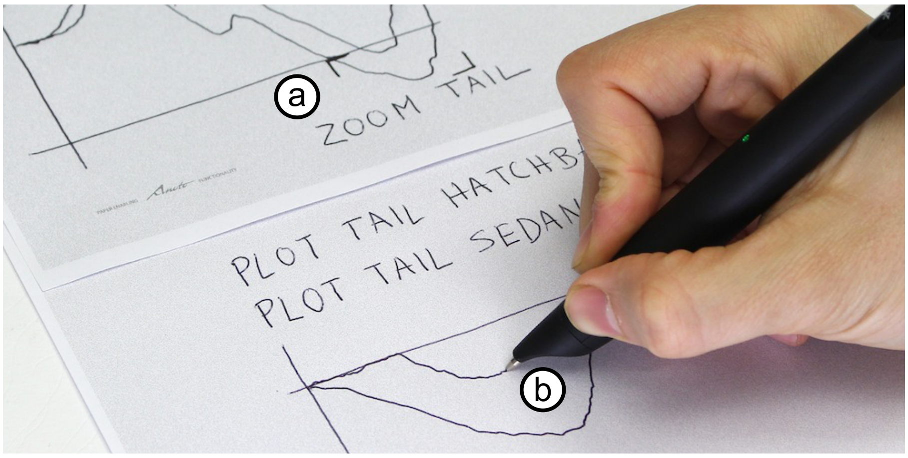

Figure 113: The user inspects a region of the wind speed charts by zooming.

Scale widget: Finally, the user wonders whether the improved turbulences will really manifest themselves in lower wind resistance; hence better gas mileage. In Figure 114a, the user writes *"plot drag*" and (b) selects the car sketch by drawing a pigtail on it. Then the user draws a vertical line, labels the line's ends "0" and "1", and traces the line using the pen. (c) Half way in the line, the system whips the user's hand sideways, creating a tick mark on the line, representing the drag coefficient of the sedan design. The user now selects the hatchback design by drawing a pigtail into it and plots the drag coefficient of the hatchback onto the line. The hatchback's drag coefficient is indeed smaller than the sedan's, which implies that this particular hatchback design can actually be expected to offer more gas mileage that the sedan.

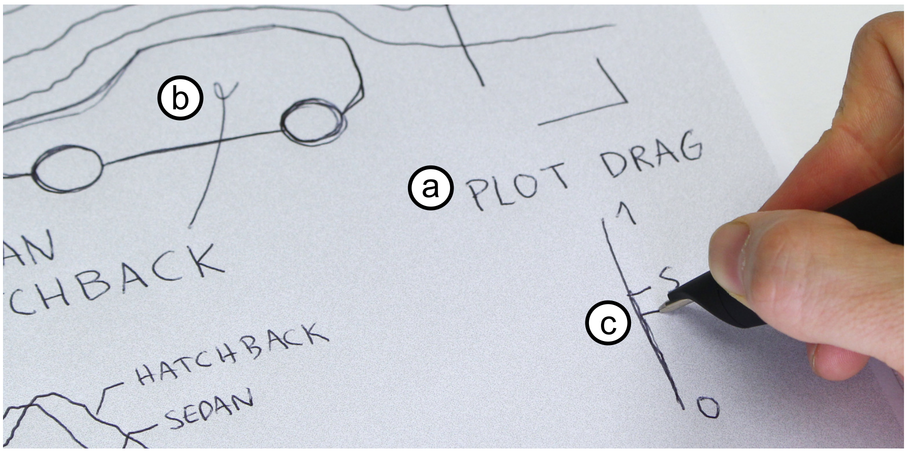

Figure 114: To obtain the drag coefficient, the user (a) writes the command and sketches a vertical line, and (b) selects the desired car with a pigtail. Then, (c) our system outputs the value as a tick-mark to the right as the user traces the line.

4.3.2 More application scenarios

As discussed earlier, we think of muscle-plotter as a tool for sensemaking activities. To emphasize this point, here are 5 other scenarios we have enabled using muscle-plotter:

Application 2: Solving Mathematical Exercises. Figure 115 depicts how a user interacts with *Octave* (software running in the backend) to solve mathematical problems through pen and paper.

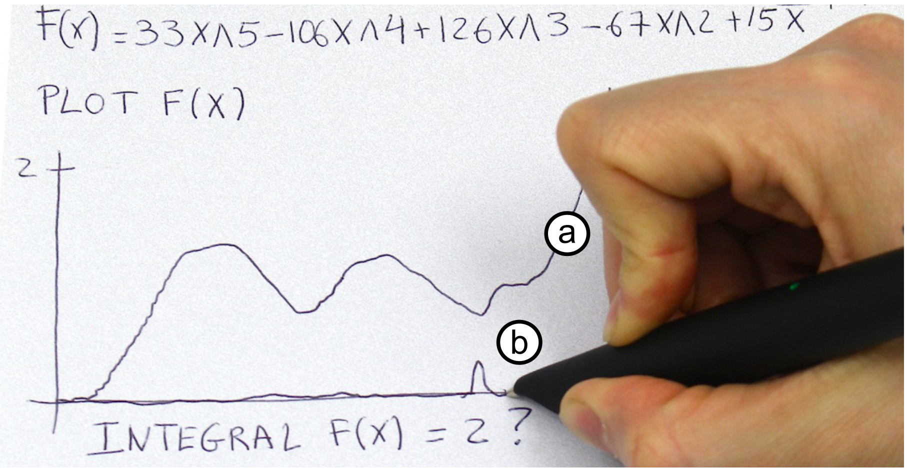

Figure 115: (a) Plotting a fifth degree polynomial and (b) finding out via a query where the integral of this polynomial equals 2 units.

In Figure 115a, the user first plots a fifth degree polynomial. Using the notation "*f(x)=…*" the polynomial is saved for future reuse as f(x). Then (b) the user queries for the point in which the integral of *f(x)* totals 2 by writing "*integral f(x)=2*" followed by a question mark. The answer is given as a tick mark while the user traces the X-axis.

Application 3: RC Circuit Simulator. Figure 116 depicts our simple filter design application built around first-order RC filters.

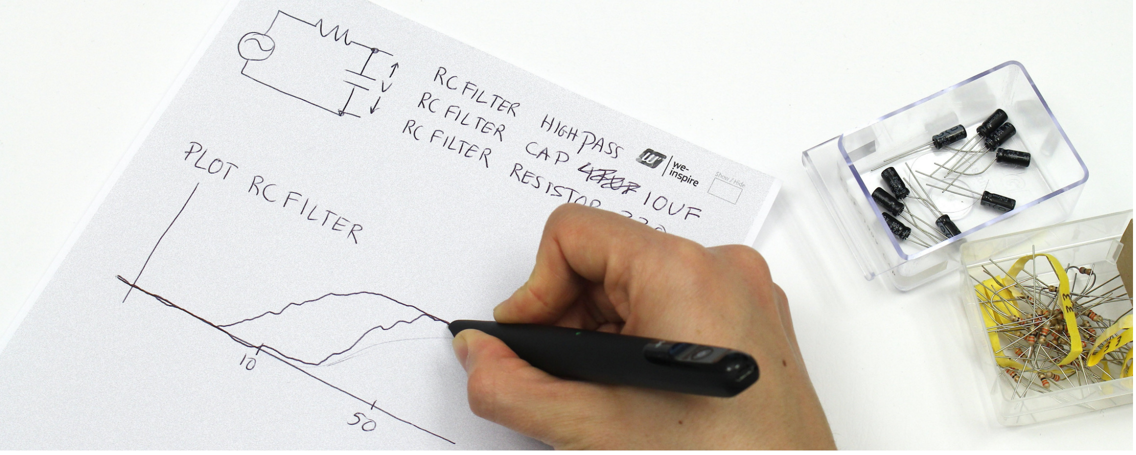

Figure 116: Iterating through the design of a RC filter by scribbling the old value a user can input a new capacitor value and then re-plot the filter response.

Figure 116 shows a user iteratively exploring different filter designs by observing their frequency response. The user first defines the R (resistance) and C (capacitance) values and the filter type and plots the filter's frequency response using muscle-plotter's line chart. (a) Unhappy with the filter, the user scribbles the capacitor value and writes a new one. (b) Muscle-plotter re-computes the frequency response and outputs it to the user in the line chart. The new filter now high-passes around 50 Hz.

Application 4: Interacting with Forms. Now we demonstrate a user interacting with a simple form in order to configure muscle-plotter's stimulation parameters.

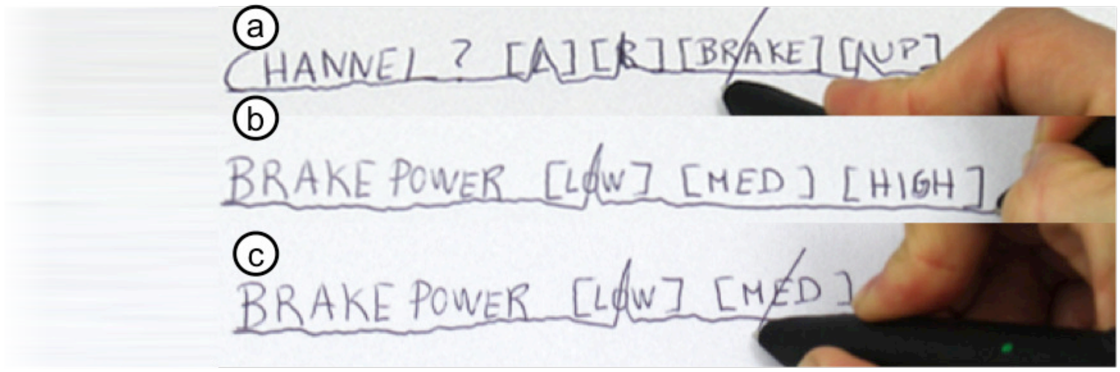

Figure 117: Simple form widgets for quick queries in muscle-plotter: (a) a multiple checkbox and (b) a radio button.

In Figure 117a, the user queries which channels are active by writing "*channel*" followed by "*?*", then creates a list of channel names (each in square brackets). While the user traces the options, muscle-plotter outputs a tick mark on those that are active. The user decides to activate the brake channel, by crossing its checkbox. Now, in Figure 117b, the user repeats the same procedure on a radio button to adjust the intensity of the "brake" stimulation. Lastly, in Figure 117c, the user adjusts the power of the "brake" to medium.

Application 5: Fitting a Trend Line in Statistics. Figure 118 depicts our simple statistics application built around muscle-plotter.

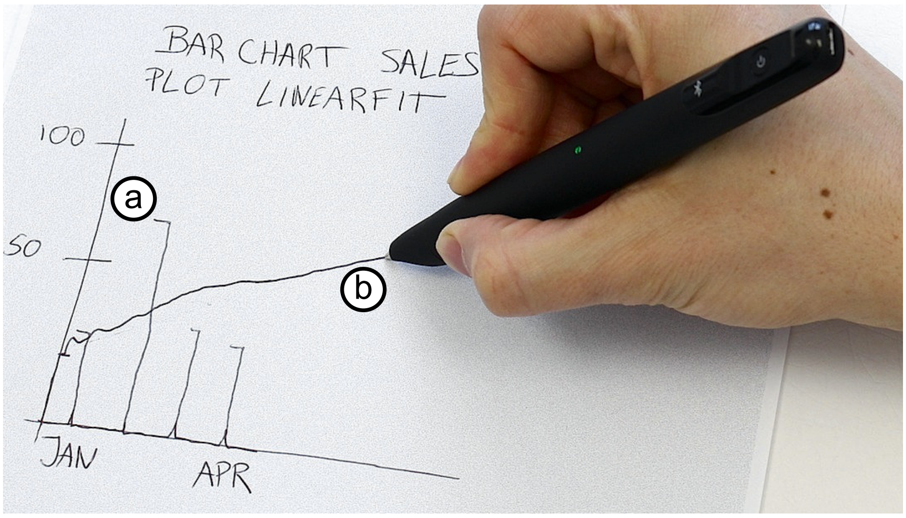

Figure 118: Fitting a linear regression in a bar chart.

In Figure 118, a user is exploring a dataset with sales arranged by months. The user does so by plotting the sales in a bar chart. This widget behaves similar to the line chart, except that muscle-plotter outputs the tick-marks for the horizontal axis as the user scans the axis. In Figure 118a, the user scans vertically from each tick-mark and muscle-plotter outputs a dash to the left and a pen up at the end of each bar. Lastly, in Figure 118b, the user makes a prediction of the next month sales by fitting a linear regression through the plot, which muscle-plotter outputs as a line chart.

Application 6: Optical Lenses Simulator. Figure 119 depicts our simple optics application that allows exploring how rays of light refract in convex and concave lenses.

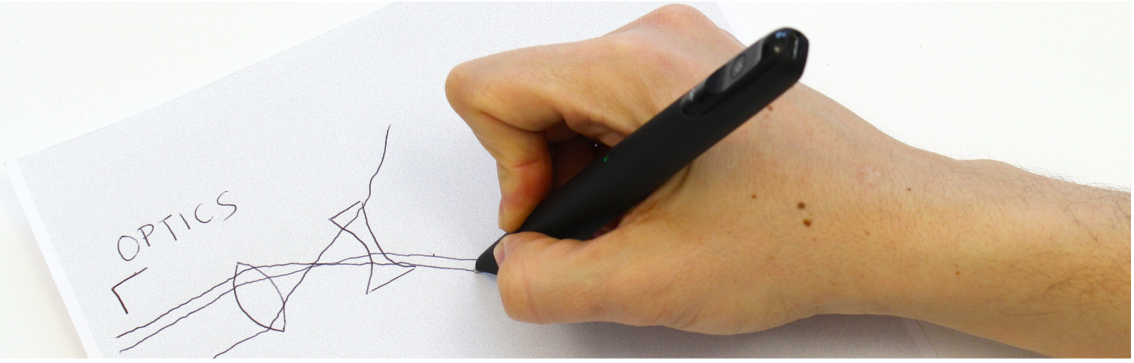

Figure 119: Using muscle-plotter to explore how rays of light refract through convex and concave lenses.

In Figure 119, a user sketches a convex lens and explores how the light rays refract through it. Muscle-plotter outputs this by deviating the ray in a slope. Now the user wonders about how to further defocus the resulting beam and draws a concave lens to explore its refraction properties.

4.3.3 Summary of widgets

We just saw six examples of interacting with muscle-plotter, each of which was implemented in the form of a widget. Our widgets leverage pen-input techniques such as crossing to select 1D primitives (*CrossY* [5]), underline and crop marks (*Papiercraft* [96]), and a pigtail gesture to select 2D primitives (*Scriboli* [63]). All widgets output when the user crosses into their boundaries. The output ends when muscle-plotter stimulates the muscles that cause the user to lift the pen's tip from the paper. We now summarize the set of widgets:

Line chart: The walkthrough contained three specialized instances of the line chart widget, i.e., wind tunnel, the tail profile plots and the zoomed plots. The line chart widget supports the following interactions: zoom in/out by relabeling the axes, zoom in by selecting with crop marks, fitting a line through existing data points, performing operations on data traces and re-plotting the results as a trace (e.g., derivative of a function). Also, the chart widget supports charts that do not start at the origin.

Scale Widget: The scale widget outputs a single value per interaction, which is useful for comparing values. In case of clustered values in the same scale, muscle-plotter will output them by alternating the tick-marks to the left/right. In the walkthrough and scenarios, we featured the scale widget in: finding the drag-coefficient, finding an integral of a function and plotting individual bars in a chart. Just as with the axes of a plot, the scale widget allows to redefine its axis for zoom in/out. The scale widget is also used to allow users to find whether a plot starts at the origin or crosses the Y-axis. For this we insert a scale widget in the line chart's Y-axis, which the user scans to find a tick-mark representing the plot's starting point (e.g., in Figure 112).

Radio button: The radio button allows selecting one option out of a range or receiving output from the option that is active. This can also be used for a yes/no dialog, which is useful when asking simple questions to muscle-plotter (e.g., to check if a number is prime). In the examples above, we demonstrated this widget at the example of a user configuring the intensity of an EMS channel.

Checkbox: The checkbox is an extension of the radio button that allows for multiple choices to be active. This widget is useful for finding elements in lists such as options in a combo box. We demonstrated it at the example of a user querying which EMS channels are currently active.

4.3.4 Contributions, benefits, and limitations

Our main contribution with muscle-plotter is that it takes electrical muscle stimulation interfaces to the next level by demonstrating interactions that render significantly more complex data than previous EMS-based systems.

To achieve this, the key idea behind Muscle Plotter is to make the user's hand sweep an area on which muscle-plotter renders curves, i.e., series of values, resulting in more information than that conveyed by a single actuated pose. By allowing EMS to produce spatial output, muscle-plotter enables a range of sensemaking activities, as we illustrated at the example of several scenarios, including the aerodynamics scenario at the beginning of our paper.

In the process of creating muscle-plotter, we developed a number of EMS-related techniques. (1) To quickly actuate the user's wrist (instead of merely informing the user where and when to turn), we simultaneously actuate pairs of *opposing* wrist muscles, resulting in a more controlled motion. (2) We compensate for latency introduced by the Anoto tracking system (around 90 ms) by extrapolating pen positions. (3) In order to turn around sharper angles, we slow down the user's wrist. We achieve this by actuating the *flexor carpi ulnaris* muscle. This causes this muscle to push the pen against the paper, increasing friction, and slowing the hand down. (4) To mark the end of an output trace, we actuate the user's wrist upwards, hence lifting the pen's tip away from the paper.

On the flipside, like any solution based on electrical muscle stimulation, muscle-plotter requires placing electrodes, calibration and recalibration if the muscles fatigue over time [73]. Furthermore, since we actuate the user's wrist only (rather than wrist and shoulder) our system is limited in terms of the shapes it can output through the user's hand.

4.3.5 Implementation details

The muscle-plotter system is comprised of a wireless Anoto digitizer pen & paper [150] for input, as well as a HASOMED medically compliant 8-channel portable EMS stimulator [70] for output.

Muscle-plotter utilizes 4 channels of the EMS stimulator at 200 Hz. This EMS stimulator is powered using a battery and interfaces with muscle-plotter via USB. This allows our prototype to run based on a laptop computer with a USB Bluetooth 4.0 dongle and a USB connection. While our current version is merely portable, a signal generator such as the one we created for pose-IO could make muscle-plotter wearable.

4.3.5.1 Electrode placement

Muscle-plotter controls four muscle groups, each responsible for a different axis of motion as depicted in Figure 120: (a) The *extensor carpi radialis brevis* and partially the *flexor digitorum* move the pen to the right (assuming a right-handed user), while the *extensor carpi radialis longus* lifts the pen tip away from the paper. (b) The *flexor carpi radialis* moves the pen to the left, while the *flexor carpi ulnaris* pushes the pen down into the paper, acting as a "brake" to prevent oscillations or improve straight lines.

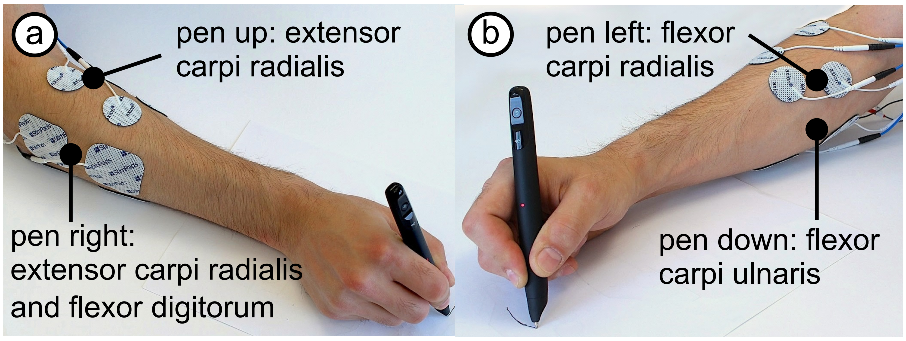

Figure 120: Electrode placement to perform (a) pen right, pen up, (b) pen left, and pen down (i.e., "brake") movements.

4.3.5.2 Calibration procedure

Muscle-plotter uses a three-step calibration procedure:

1. Comfortable use: By slowly increasing the intensity, we first determine the minimum value required to actuate the user's muscles. Then, we find the maximum intensity (in mA) and pulse width (in μs) that feel comfortable and pain-free.

2. Drawing slanted lines: Users start by trying to draw a straight line onto the paper. At a random position along the way, muscle-plotter actuates their wrist, turning the remainder of the user's line into a slanted line. By performing this step repeatedly with varying pulse widths, the calibration routine determines the pulse widths that produce 20, 40 and 60-degree slants to the left and to the right.

3. Drawing straight lines: Users draw slanted lines, as in the previous step. When they reached a second random point along the way, the system actuates them to draw a straight line. By performing this step repeatedly with varying pulse widths, the calibration routine determines two pulse widths that return the user's hand to drawing a straight line.

Figure 121 depicts the average pulse widths determined by means of this calibration routine for all users in our experiment. Similarly, the calibration routine determined that the extensor muscle should be actuated using an average intensity of 10.3 mA (SD = 1.3 mA) and the flexor muscle using an average of 8 mA (SD = 0.5 mA).

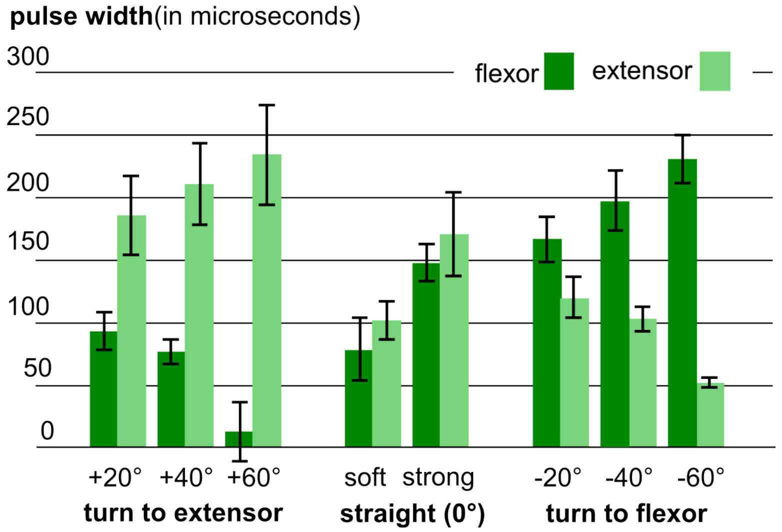

Figure 121: Average pulse width per slant, for both extensor and flexor muscles, for all participants of our experiment.

4.3.5.3 Control loop and modelling the human wrist under EMS actuation

The purpose of muscle-plotter's control loop is to actuate the wrist to reach a target as quickly as possible, yet without overshooting.

The main algorithm receives the tracking data asynchronously (i.e., the X/Y location of the Anoto pen in coordinate system of the paper). It then re-evaluates the situation and picks one of three possible strategies: aim for a target, compensate if the user is constantly lagging behind the target trace, or brake. The loop executes this strategy by sending electrical impulses to the user's wrist muscles. In the following, we look into the each of these steps in detail.

1. Receiving pen location data: The communication between the Anoto Windows API (from the vendor *we'inspire*) and muscle-plotter is done via Open Sound Control (OSC). Each OSC packet is sent to muscle-plotter's Python server. It encapsulates an X/Y position, an event type (pen up, pen down or pen drag), and a timestamp. We found the latency of this operation to be below 1 ms for both loopback and Ethernet.

2. Evaluating the current strategy: Our system aims for the most likely target, which may not necessarily be the closest one. It determines the most likely target by extrapolating the current location based on the current speed, as depicted in Figure 122. This approach allows the system to cope with differences in hand movement velocities and with the Anoto tracking latency (around 90 ms, determined using a high-speed camera).

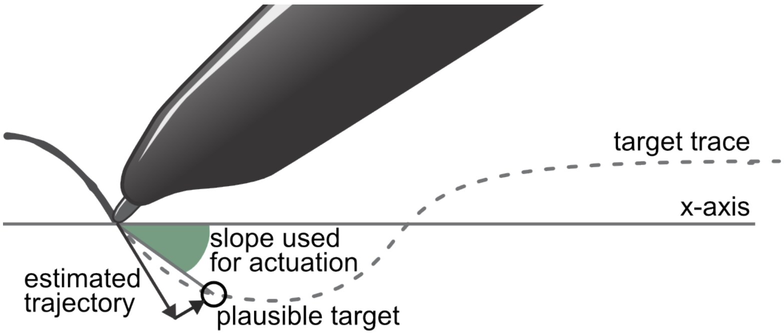

Figure 122: Key principles in muscle-plotter's control loop.

Then, for every decision cycle the system evaluates if the user is: (a) close to a plausible target or (b) consistently lagging behind the target trace or (c) approaching the target at a high speed with a chance of overshooting. Accordingly, it performs one of the following three responses.

Strategy (a): aim for a target. In order to reach a target, muscle-plotter computes the slant between the target and the current estimated location (Figure 122). Then, according to this slant, muscle-plotter finds a pair of pulse widths that is the closest to the pre-calibrated slants from the user's calibration dataset. Muscle-plotter actuates the wrist by applying a simultaneous signal to *both* sides of the wrist. We found this to allow for smoother control than when actuating just one side at a time. This is particularly important if the user's dragging speed is fast.

Strategy (b): compensate for wrist lagging behind. If the accumulated error (i.e., sum of all distances to previous targets) exceeds 20 mm we consider that the user is lagging behind. To compensate, we dynamically increase the intensity of the muscle that moves the pen closer to the target in 1 mA steps (up to 2 mA). Finally, we reset the accumulated error and disable the intensity boost when the user's trace crosses again with the target trace.

Strategy (c): brake. If the user approaches a target too fast, there's a risk of overshooting. If the system observes the user's hand approaching the target at increasing speed and if is less than 3.3 mm away, the system "brakes" the user's wrist. By actuating the muscles that push the wrist into the paper, the system increases the friction between pen and paper, hence slowing the pen's tip down. The brake strategy is also particularly useful for improving the performance of straight lines and to prevent undesired oscillations.

3. Executing the current strategy: Muscle-plotter executes one of the strategies by sending the electrical impulses to the user's wrist. This is done in a separate thread, which allows it to keep a constant stimulation frequency of 200 Hz. To achieve precise control, muscle-plotter controls the pulse width (from 20 µs to 500 µs) of the waveform rather than the amplitude. Changing the amplitude is only used while the user is lagging behind, i.e., strategy (b).

Lastly, to terminate the current output trace, muscle-plotter moves the tip of the pen away from the paper by actuating the user’s wrist upwards.

4.3.5.4 Heuristics used in the control loop

To avoid muscle-plotter from overshooting the target we hand-tuned the main loop’s parameters as follows.

The 20 mm threshold is sufficient to detect when the user is lagging behind without producing oscillation effects. If this value is heavily decreased (e.g., 2 mm) the automatic compensation in intensity will create unwanted oscillations.

The 3.3 mm braking distance provides sufficient braking distance to avoid overshooting. We found that the braking technique is most efficient if initiated at least 2 mm away from the target, otherwise the system tends to overshoot.

Our brake implementation uses 11 to 44 impulses depending on the user's speed. We found that when the user approaches slowly (e.g., around 3.3 mm/s) 11 impulses are sufficient to halt the trajectory without undershooting. Intuitively, when the user’s speed increases we increase the brake impulses; we found that 44 pulses allows us to brake before the target (no overshoot) at speeds up to 10 cm/s.

4.3.5.5 Computing geometry for widget interaction

The geometry computed in muscle-plotter uses OpenCV via Python bindings.

For instance, line chart widgets start when the user draws two axes that intersect one another. This is calculated as the cross product of two fitted lines between the two drawn axes. Likewise, the two crop marks that define the wind tunnel widget are detected using the same method for each; the perpendicular lines determine the wind tunnel dimensions. Lastly, for the cross-section and scale widget, the user drawn lines are fitted with a line.

4.3.5.6 Handwriting recognition using Tesseract

To recognize hand-written input, we integrated *Tesseract* [115], a trained recognizer with several languages and one of the most robust open-source optical character recognizers (OCR) [142]. Our system loads only the English language and common symbols such as “?”, “(“, etc. Before applying Tesseract to the user’s strokes we: (1) concatenate all strokes that are less than 3 mm apart from their center (enables multi-stroke writing). (2) Interpolate the points from the Anoto tracking to complete a line. (3) Convert to a black and white image for OCR. (4) If the next stroke is more than 1 cm apart, the previous strokes are grouped into a command keyword and evaluated with the OCR. (5) After a command keyword is detected, we evaluate the incoming strokes character by character; this allows to directly write commands such as “*PLOT F(X)=SIN(X)*”.

Once Tesseract returns the recognized text to muscle-plotter, our system replaces it with the word from its custom keyword dictionary that is closest in terms of Levenshtein distance (edit distance). This corrects for common misrecognitions such as “L” for “(“ or “X” for “K”.

4.3.5.7 Application-specific implementations

We now detail the implementations that are specific to each of our applications.

Mathematical formulae manipulation & solvers. To solve mathematical formulas, we invoke Octave by means of its Python bindings. We use it for mathematics (derivatives, integrals, and so forth) and for plotting functions from formulae. We interface muscle-plotter and Octave by: (1) converting user-notation to Octave notation. When the user writes “*sin(x)*”, for example, we convert it to “*feval((sin(x), <range-of-plot>)*”; then, (2) send the formulae to Octave, sample it into points and return it to muscle-plotter; and, lastly, (3) transform these points to the Anoto paper’s coordinates and into the user-defined axis.

RC-Filter response. We implemented a simple solver for high and low pass RC circuit filters. It works by solving the filter equations directly in the frequency domain. The values for R and C are read from the users’ input when they write “*cap 10UF*” (i.e., 10 µF) and “*resistor 330*” (in Ohms).

Statistics example. To perform statistical operations we use Python’s *Scipy* Statistics package. This allows us to compute standard deviations and regressions, which we use in a simple demo application that draws a linear regression through bar charts.

Optical lenses ray casting. The optics demonstration is a simple 2D ray casting based on [92] that deals exclusively with concave and convex lenses. We disambiguate the lens type by measuring the width of the lens’ middle section against a fixed threshold (i.e., convex is wider).

Wind tunnel simulator. The wind tunnel simulation is based on the Lattice-Boltzmann equations and adapted from Schroeder’s implementation[139]. To run a wind tunnel simulation (i.e., to compute the wind speed streamlines) we: (1) rotate the model according to the rotation of the wind tunnel boundaries; (2) extract the shapes drawn inside; (3) down-sample these shapes (e.g., the car) into a binary matrix of 200 px height; this matrix contains the obstacles to the wind flow; (4) for 30 steps we execute Lattice Boltzmann by advancing the streamlines one step at a time and re-evaluating the collisions to compute the velocity flow; (5) once the user draws a line, we use Python’s *Matplotlib* streamlines function to obtain the streamline that starts at the user’s pen-down position.

4.3.6 User study: validating our implementation

To validate the mechanism behind muscle-plotter, i.e., the production of spatial output using EMS, we conducted a user study. Our focus was on which types of signals can muscle-plotter reproduce *as is* and which signals require *pre-processing*. In the study, muscle-plotter actuated participants to repeatedly draw curves and we measured how closely these curves matched the intended target signal.

4.3.6.1 Task

For each trial, participants plotted one function onto paper using muscle-plotter as depicted on Figure 123.

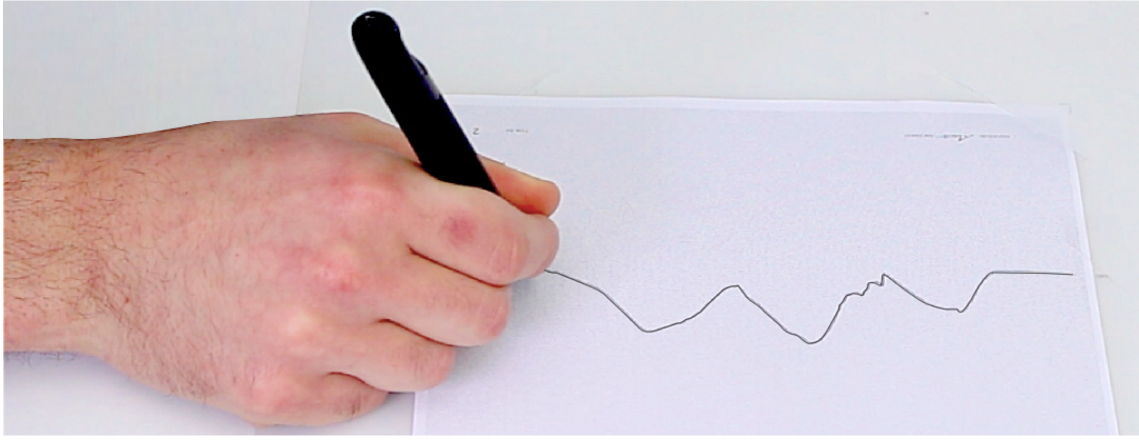

Figure 123: One trial: one plot (image from the study).

In each trial, participants plotted one of eight mathematical functions. Participants placed their hand on the left side of the paper so as to point the pen at the marked starting position. Participants then moved their hands towards the opposite side of the page, at any speed they chose. During this period muscle-plotter actuated their wrist, hence plotting the curve. When participants reached 16 cm from the left to right margin, the trial ended. The system recorded the intended function, the user drawn function, as well as the time between start and end of plot.

Every participant performed each function twice, resulting in 128 trials: 8 functions × 2 repetitions × 8 participants.

4.3.6.2 The dataset

The eight mathematical functions are shown in Figure 125. The first six functions were composed by adding off-phased sine waves of increasing difficulty; hence we denote them as *Sin1*-*Sin6* (ascending in frequency and amplitude). We added a triangular wave (denoted as *Tri*) and a sine wave that ended in a flat section (denoted as *Flat*) to explore how the system behaves with regards to abrupt changes in slope and curvature (as well as to prevent participants from getting used to sine wave patterns).

4.3.6.3 Apparatus

Figure 28 shows our apparatus. Participants wore muscle-plotter's electrodes on the wrist flexor and extensor muscles (as described in Implementation). They were seated with the dominant forearm rested on the table to reduce fatigue.

We used the EMS control loop described in "Implementation details", which actuated flexor and extensor muscles simultaneously but without the brake channel, which was introduced as an outcome of this study. The muscle-plotter software administered the respective functions to the user; all other functionality was disabled. Muscle-plotter was calibrated with the procedure described in the Implementation section.

Figure 124: Setup for our experiment (image taken from the study).

4.3.6.4 Participants

We recruited 8 right-handed participants (1 female, M = 23.9 years old) from our local university. With consent of the participants, we videotaped the study sessions.

4.3.6.5 Results

Raw data: Figure 125 shows the curves drawn by participants. The average error from target across all 128 trials was 4.07 mm (SD = 3.03 mm).

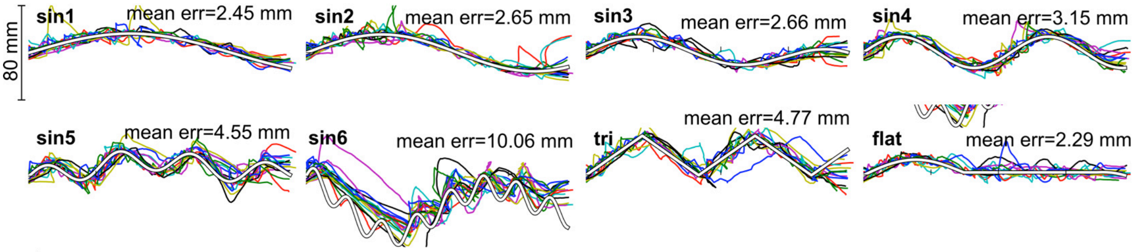

Figure 125: Raw data from 8 participants and 128 trials. The white trace represents the desired trajectory and colored traces depict user drawn plots (scale in mm). Average per function represents error (distance to target).

Preservation of sine-based functions: Figure 125 also shows the average error per by function (Sin1: M = 2.45 mm, SD = 1.42 mm; Sin2: M = 2.65 mm, SD = 1.29 mm; Sin3: M = 2.66 mm, SD = 0.90 mm; Sin4: M = 3.15 mm, SD = 1.00 mm; Sin5: M = 4.55 mm, SD = 1.31 mm; Sin6: M = 10.06, SD = 2.67 mm). As expected, there was an increase in error with the increase in the function's highest partial confirmed by a linear regression with $R^2 = 0.64$ through Sin1 – Sin6. To provide an estimate of how much each trial differed between each other, the reported standard deviations (SD) are between the mean errors of each trial.

Preservation of non-sine functions: When plotting the two functions that contained sharper changes in slope and curvature participants performed similarly to the sine waves (*Tri*: M = 4.77 mm, SD = 1.53 mm; *Flat*: M = 2.29 mm, SD = 0.82 mm).

Preservation of features: The plots in Figure 126 illustrate in how far plots made through muscle-plotter preserved the original function. The plots show frequency histograms produced by means of Fourier transformation. We see the original signal in green, as well as user-specific jitter, i.e., noise, in red. As the plots illustrate, jitter tends to revolve around wavelengths smaller than 0.5 cycles/cm.

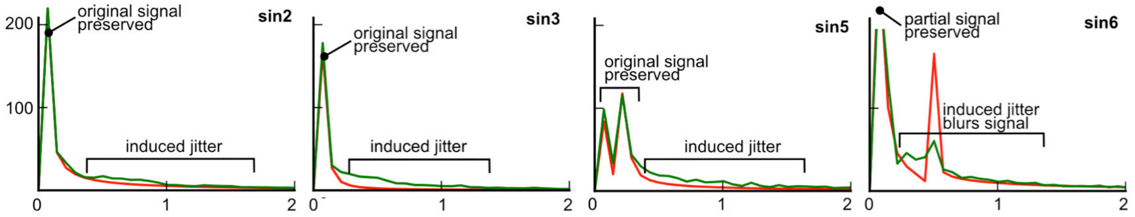

Figure 126: Fourier transforms: original signal (red) and users output (green).

Jitter in Sin2 and Sin3 has very little impact on the signal (the same as observed in Sin1 and Sin4). This gets more challenging with Sin5, which has its highest partial at a wavelength of around 0.23 cycles/cm. Still, signal and jitter are clearly distinct, which means that the original function still stands out clearly, so that awareness of one's jitter allows users to visually filter out the noise. The distance between signal and jitter gets smaller with increasing signal frequency until they start to overlap in Sin6, suggesting that part of this signal has drowned in the noise and thus has become unrecognizable.

We conclude that muscle-plotter is suitable for reproducing signals of up to 0.3 cycles/cm wavelengths, but should not be used for frequencies higher than this.

Speed/Accuracy tradeoff: The fact that participants picked their own pace resulted in a wide range of speeds. Figure 127 illustrates the resulting speed/accuracy tradeoff. Participants plotted the 16 cm-wide functions in between 7.98 s and 29.5 s (M=16.17 s, SD=4.90 s).

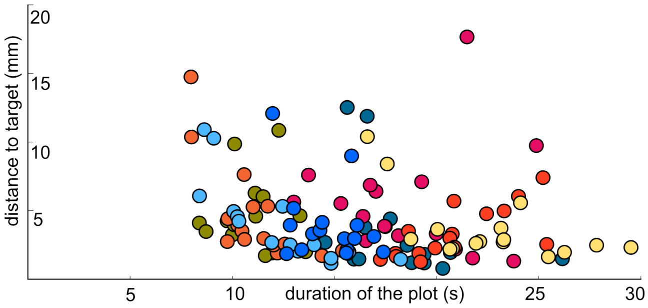

Figure 127: Average error (in mm) for each trial in relation to its duration (in seconds). Each color is a different participant.

4.3.6.6 Discussion

The main finding of the study is that muscle-plotter is able to reproduce signals up to 0.3 cycles/cm wavelengths (e.g., Sin5 signal), while higher frequencies (e.g., Sin6 signal) cannot be reproduced as is. This insight is useful in that it allows application designers to decide how to pre-process their data before outputting it through muscle-plotter, e.g., by stretching the signal horizontally before plotting. Interestingly, two out of the eight participants stated that they had not looked at the paper while plotting. This happened naturally, since we had not instructed them either way.

We opted to study the performance of rhythmic curves (i.e., based on the arithmetic combination of multiple sine-waves) as these allowed us to separate out signal and jitter using the frequency analysis we presented in Figure 126. However, in our experiments we found that non-rhythmic curves behaved similarly. The reason is that muscle-plotter plots at a slow enough pace that every movement is induced separately. In fact, muscle-plotter moves the wrist from target to target, rather than in a rhythmic manner. Hence, the way in which our system plots is different from the way humans tend to sketch waves quickly by performing a single oscillating movement.

Based on these results we improved the system by adding the aforementioned *brake* functionality, i.e., muscle-plotter slows down the user by actuating the wrist muscles downwards as to push into the paper; this increases the friction between pen and paper. Also, this helps returning the wrist to a neutral position, improving the quality of straight lines.

4.3.7 Conclusions on muscle-plotter

We described muscle-plotter, an interactive system based on electrical muscle stimulation that provides pen-on-paper interactions for both input and output. In our system, users input by writing, e.g., writing mathematical formulae or drawing shapes. The system outputs by actuating the users' wrist so as to draw graphs, strokes, etc.

We designed muscle-plotter to render significantly more complex data than previous EMS-based systems. The key idea behind muscle-plotter is to make the user's hand sweep an area on which muscle-plotter renders curves, i.e., series of values, and to *persist* this EMS output by means of a pen. This allows the system to build up a larger whole, enabling it to assist users in cognitively demanding activities, such as designing an aerodynamically sound vehicle, by providing users with access to a computer system while they are sketching on pen and paper. Still, the use of EMS allows muscle-plotter to achieve a compact and mobile form factor.

4.4 Discussion on information access using EMS

Using these last three prototypes we established that proprioception is suitable as an interaction modality. We now discuss some of the open questions that our proposal of proprioceptive information exchange has raised.

4.4.1 Implementing proprioceptive interaction with alternative actuation hardware

To implement proprioceptive interaction in a device, one needs to augment the target device with capabilities to read the user's movements (or muscle activity) and to actuate the user's limbs. While we solved the latter by using EMS (since it lends itself well to create wearable devices) we believe that the idea of proprioceptive interaction is a broader concept that is agnostic to the hardware implementation. We could envision devices similar to pose-IO or muscle-plotter but, instead, are realized via mechanical actuators.

4.4.2 Limited bandwidth

Our examples of eyes-free interaction suggest that this modality is especially relevant when users cannot rely on their visual apparatus, the reason being that the human proprioceptive system is not trained for information access (since it is tuned to perform semi-unconscious tasks such as standing upright) nor able to process high volumes of information, i.e., its bandwidth is certainly lower than the visual apparatus. This is why most of our examples fall under the category of microinteractions [7]. To demonstrate more meaningful interactions, in muscle-plotter, we added a pen to the user's hand to persist the EMS output—turning it into a sort of "paper display".

4.4.3 Beyond hands and upper limbs

We believe that the concept of proprioceptive interaction should not be restricted to the hands. In Figure 128 we envision two proprioceptive interaction designs, one for the upper arm (using the *biceps* for flexion) and one for the feet (using the *gastrocnemius* for plantar flexion). These additional possibilities open up interaction spaces for scenarios where the hands are occupied with another task (eyes-free + hands-free).

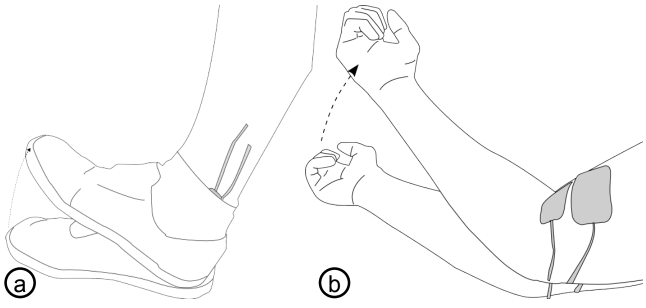

Figure 128: Extending proprioceptive interaction to other parts of the body: (a) the feet and (b) the upper arm.

4.4.4 Information access vs. learning a new skill

Finally, while we designed affordance++ to transfer information between object and user, it is also a tutorial system—users learn how to operate an object by doing it, rather than watching a video or reading a manual. This "learn by doing" echoes the theory of embodied learning, which emphasizes the value of haptic experiences in learning [102, 140]; we see enormous potential in EMS for e-learning physical skills.

5 CONCLUSIONS & DISCUSSION

In this chapter, we digest the work presented so far by summarizing and discussing the benefits of interactive systems based on EMS. Then, we close by sketching an outlook for the field, focusing on open questions and technical challenges.

5.1 Benefits of interactive systems based on EMS

We start by summarizing and discussing the three benefits that we believe make interactive systems based on EMS worth exploring.

5.1.1 Miniaturization of force feedback

One of the key benefits of interactive systems based on EMS is that they miniaturize well— EMS offers serious potential for building wearable interfaces, an advantage we found and explored in our first line of work (i.e., increasing realism in VR/AR).

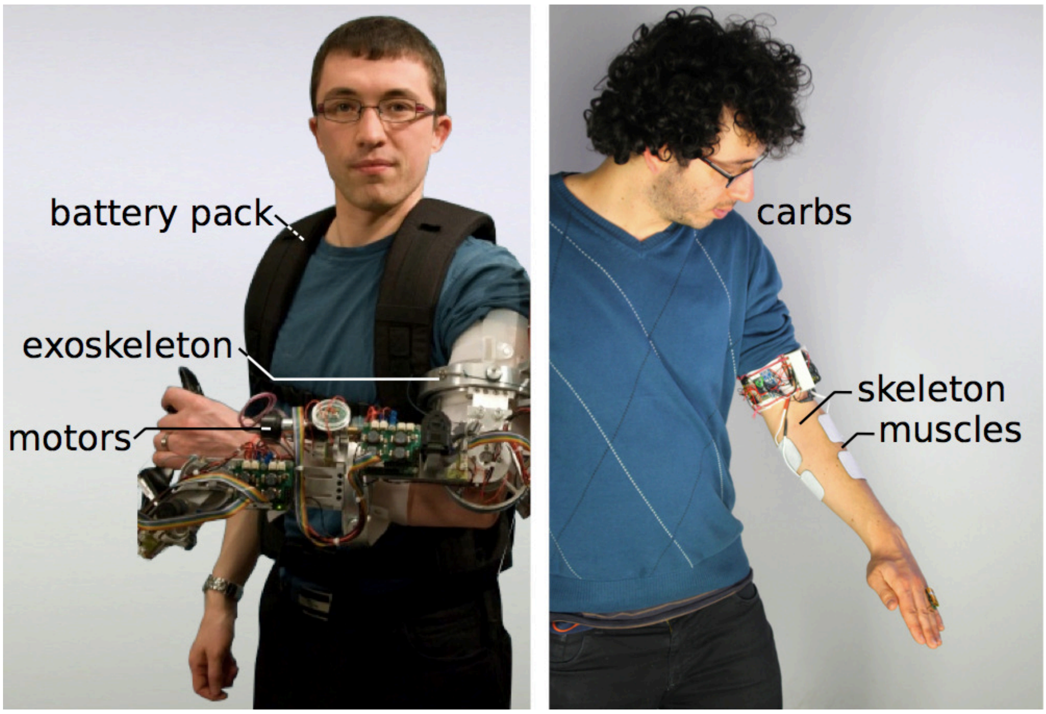

Figure 129: (left) While devices based on mechanical actuation *add* mechanical components to the user’s body, (right) systems based on electrical muscle stimulation instead *borrow* the user’s skeleton and muscles (Exoskeleton image source: Université Libre de Bruxelles; used with permission.)

This particular advantage of EMS systems becomes more evident when we compare it with more traditional approaches, such as motorized exoskeletons. While exoskeletons tend to enclose the human body with mechanical structures (Figure 129a), EMS-based systems achieve a similar effect without the bulky hardware (Figure 129b). This is the result of a single central and unique aspect of EMS systems: where mechanical solutions *add* mechanical components to the user's body, EMS-based devices instead *borrow* "components" from the user, in particular the "mechanics" already contained in the human body. This makes EMS systems comparably small and even allows them to be worn—invisibly—under the user's clothes.

5.1.2 Next step after wearables: devices becoming more personal

This borrowing of parts of the user's body not only allows for a small form factor, but also results in a particularly intimate relationship between user and device, which was something that we observed in user's comments. For instance, participants of our user studies voiced the feeling of a blurred ownership of one's own hand (in pose-IO), or attributed external agency to an inanimate spray can that was, in fact, actuated by means of their own muscles (in affordance++). We believe this ability to alter agency is a special property of EMS since it *actuates the human body from within*. However, further investigation with controlled experiments is needed to arrive at precise conclusions.

Furthermore, recalling the history of interactive computing with which we opened this dissertation with, we can now wonder whether devices that borrow the user's body might in fact be the next significant device paradigm.

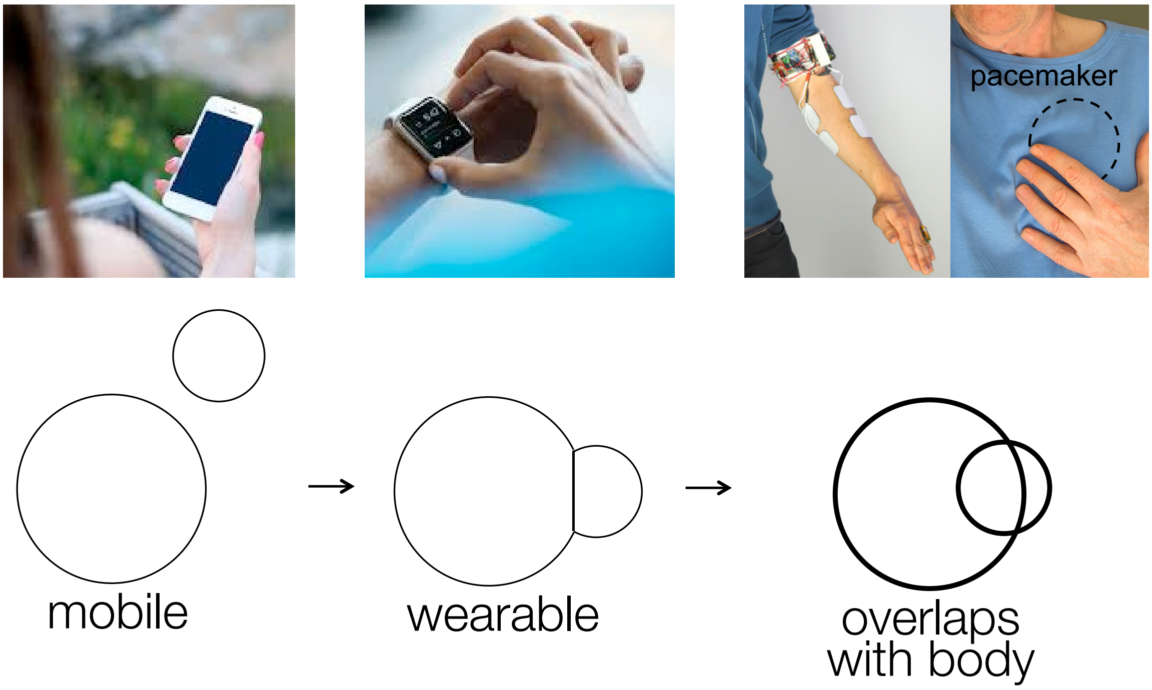

Figure 130: A possible evolution of interactive devices. (Images in CC0 Creative Commons by *JESHOOTScom*, *Free-Photos* and *geralt_*)

Figure 130 illustrates this by sketching out a possible evolution of interactive device paradigms. By considering our proposal, *devices that overlap with the body*, as a natural consequence of wearable interfaces it is possible to consolidate several lines of research in which devices literally share the user's body, such as implanted interfaces [66, 95], on-body input [143, 56, 134], and interactive systems based on EMS.

5.1.3 Bi-directional communication channel to the user

Traditional interactive devices, including mobile devices, are separated from the user's body. Therefore, these devices do not have a continuous communication channel to the user. Wearable devices, in contrast, tend to be in physical contact with the user's body. This allows them to send messages to the user, e.g., by stimulating the user's skin using vibrotactile actuators. Implanted devices [66, 95] are located inside the user's body, where they are able to continuously read and actuate upon some of the user's internal properties—as insulin pumps or pacemakers do—but their ability to talk back to the user is, unfortunately, limited.

Unlike these three categories (mobile, wearable and implanted) EMS-based devices effectively *overlap* with the user's body and this overlap allows them to maintain a continuous bidirectional channel (input and output) with the user. The result is a closed loop of input and output between user and device that users can perform eyes-free—a quality that gains in relevance as more and more wearable devices, such as smart watches, compete for the user's visual attention. While many wearable devices can notify its user using vibration patterns, EMS is substantially more expressive since a pair of EMS electrodes can communicate a continuously changing parameter to its user.

5.2 Outline and open challenges

Now, we sketch an outline for the field of interactive systems based on EMS by listing open technical and philosophical questions that we left unanswered.

5.2.1 Technical challenges

There are many exciting hardware and software challenges in EMS-based interactive systems. To overcome these limitations in the future, we, in the research community need to deal with a wide range of challenges; in particular, current limitations in EMS precision must be addressed:

1. Increase the robustness of EMS-based systems to variations in body posture and to muscular fatigue. One approach might be to add additional sensors that track limb orientation (e.g., accelerometers) and physiological signals (e.g., electromyography to measure muscle activity and chemical sensors that analyze sweat composition).
2. In order to actuate with higher precision and enable more complex poses, it might be worth implanting microelectrode arrays [18] into the respective muscles.

3. To harmonize the control loops with the user's voluntary muscle contractions, we may consider creating EMS systems that, simultaneously, sense muscle tension and thus track users' voluntary motion.
4. Concurrently with these developments, one would expect to see a series of other improvements that make EMS research easier and its outcome more applicable, such as automatic calibration procedures and methods that simplify the placement of the electrodes, embed them into textiles, and so on. Furthermore, recently, we have started to see some early examples of automatic calibration of EMS devices [84, 44, 127, 79].

5.2.2 Conceptual, philosophical & societal challenges

Resource conflicts. The idea that a device can "borrow" the user's limbs poses a number of interesting design challenges, such as the potential "resource conflict" that arises when user and device try to actuate the same limb at the same time. Our EMS-based systems solved this issue by having the EMS actuation configured to be weak enough to allow users to simply overpower the actuation. Another approach is to sense the user's voluntary motion and to abort any device action whenever the user is trying to act (e.g., as in pose-IO).

Shift in agency. Another challenge present in using any form of automated system, be it EMS-driven or motor-based, is a shift in agency. While the naïve assumption might be that any system that automatically actuates a user decreases the perceived agency, during our experiments, we observed users' comments during that suggested blurred forms of agency—hinting that the notion of loss vs. gain of agency might not be strictly binary, but rather affords "sharing agency" with a device.

We believe the question of agency is the fundamental piece for hybrid human-machine systems and the nature of EMS simply takes this question to the spotlight. We believe further investigation is required to understand how exactly is the sense of agency affected by such systems and, also, does EMS affect agency in any particular way?

This question unfolds into a series of subsequent questions, such as: do we feel like we made a choice when we find our bodies act out such a choice? Does it make a difference whether we find our body actuated by mechanical contraption, or by a more "internal" EMS device? Answering these questions could potentially revolutionize a wide range of applications designed to teach us physical skills, such as haptic tutorial systems.

Ethical repercussions of a shared agency. The consequence of systems with shared models of agency is that they disrupt the fabric of societal norms. One example at the contemporary spotlight is the moral and liability paradoxes in self-driving cars, e.g., who is liable, and morally responsible, in the event of an accident while the car drives *itself*? Similarly, when using an exoskeleton or interactive system based on EMS, we can ask an enormous amount of variations to this question. Because of the wide implications of these questions and its societal repercussions we believe these should be investigated thoroughly and openly, with as much public interaction as possible.

One powerful way to extend this discussion to the general public is to expose them to this paradox and incite critical thinking over the philosophical implications of hybrid technology that disrupts the traditional model of users controlling devices.

As an outcome of this philosophical side of our work, we created a device that reverses the traditional "the user controls the device" paradigm and, instead, controls the user. Our creation, depicted in Figure 131, is a technological *parasite* that lives off human energy: the device harvests kinetic energy by electrically stimulating participants' wrists. This causes their wrists to involuntarily turn a crank, which then powers the device *ad infinitum*.

Figure 131: Ad Infinitum, a parasitical machine that lives off human energy.

This device, which we call Ad Infinitum, was crafted to provoke users to consider the implications of this philosophical debate. This act of building a device to "ask a question", rather than to "answer one", made it more suitable for an artistic outlet. We have exhibited this device at prestigious venues, such as Ars Electronica (Linz, 2017) and Science Gallery (Dublin, 2017).

While the systems presented above are all prototypes and not commercial deployments, one cannot overlook the ethical landscape around these novel interfaces. In particular, we expect our projects to spark more discussion on the subject of ensuring that EMS interfaces are used in a safe, responsible and secure manner.

5.3 Community outreach

To allow other researchers to build on our work we have taken three steps.

1. **Open-source software.** We made the code of the following prototypes open-source:
    1. Affordance++: https://github.com/PedroLopes/affordance
    2. Muscle-plotter: https://github.com/PedroLopes/openEMSstim
    3. MR: https://github.com/PedroLopes/HoloLens-ElectricalMuscleStimulation

2. Open-source hardware. To enable researchers to get started on the—typically more time-consuming—hardware side of EMS, we created the *openEMSstim*, which is an open-source hardware module that programmatically adjusts the intensity of an off-the-shelf EMS/TENS stimulator. Our hardware design is derived from [124], which was one of the results of two workshops that we jointly conducted with Pfeiffer et al. at IEEE WorldHaptics 2015 and ACM CHI 2016 conferences. The openEMSstim's schematics, firmware code, documentation and step-by-step tutorials are available at: https://github.com/PedroLopes/openEMSstim

3. Community outreach. Finally, to make these ideas available to the youngest generation of our research community, we utilized our openEMSstim as the hardware platform for all 18 student teams of the ACM UIST 2016 Student Innovation Contest.

5.4 Final thoughts

Our eight projects were designed to spark the idea of interactive systems based on EMS, while simultaneously, to some deeper extent, working out the conceptual framework of a larger theme: devices that overlap with the user's body might be the natural successors to wearable computing.

We foresee that this theme, on the subject of biological integration of devices & human senses, will unfold as the centerpiece of Human Computer Interaction, leading to a significant fusion with the Neurosciences.

6 REFERENCES

1. Gad Alon (2013) Functional Electrical Stimulation (FES): Transforming Clinical Trials to Neuro-Rehabilitation Clinical Practice- A Forward Perspective. J Nov Physiother 3:176. doi:10.4172/2165-7025.1000176
2. Brian Amento, Will Hill, and Loren Terveen. 2002. The sound of one hand: a wrist-mounted bio-acoustic fingertip gesture interface. In CHI '02 Extended Abstracts on Human Factors in Computing Systems (CHI EA '02). ACM, New York, NY, USA, 724-725. DOI=http://dx.doi.org/10.1145/506443.506566
3. Hideyuki Ando, Takeshi Miki, Masahiko Inami, and Taro Maeda. 2002. The nail-mounted tactile display for the behavior modeling. In Proceedings of ACM SIGGRAPH 2002 conference abstracts and applications, 264-264. DOI=http://dx.doi.org/10.1145/1242073.1242277
4. Hideyuki Ando, Eisuke Kusachi, and Junji Watanabe. 2007. Nail-mounted tactile display for boundary/texture augmentation. In Proceedings of the international conference on Advances in computer entertainment technology, 292-293. DOI=http://dx.doi.org/10.1145/1255047.1255131
5. Georg Apitz and François Guimbretière. 2004. CrossY: a crossing-based drawing application. In Proceedings of the 17th annual ACM symposium on User interface software and technology (UIST '04). ACM, New York, NY, USA, 3-12. DOI: https://doi.org/10.1145/1029632.1029635
6. Myo Armband, https://www.myo.com/, last accessed on 2018/02/02.
7. Daniel Ashbrook, Enabling Mobile Microinteractions. Ph.D. Dissertation. Georgia Tech, Atlanta, GA, USA.
8. Serial-Library Axivity, http://axivity.com/downloads, last accessed on 13/03/2015.
9. N. Baker. Prism Nightlight, http://nicholas-baker.com/post/70963455726/prism-nightlight-the-goal-of-this, last accessed in 9/9/2014.
10. Amit Barde, Matt Ward, William Helton, and Mark Billinghurst (2016). A Bone Conduction Based Spatial Auditory Display as Part of a Wearable Hybrid Interface. In Proceedings of ICAD'16, International Conference on Auditory Display (ICAD)., DOI:http://hdl.handle.net/1853/56563
11. Patrick Baudisch, Henning Pohl, Stefanie Reinicke, Emilia Wittmers, Patrick Lühne, Marius Knaust, Sven Köhler, Patrick Schmidt, and Christian Holz. 2013. Imaginary reality gaming: ball games without a ball. In Proceedings of the 26th annual ACM symposium on User interface software and technology (UIST '13). ACM, New York, NY, USA, 405-410. DOI: https://doi.org/10.1145/2501988.2502012
12. Mohamed Benali-Khoudja, Moustapha Hafez, Jean-Marc Alexandre, and A. Kheddar. Tactile interfaces: a state-of-the-art survey, In Proceedings of International Symposium on Robotics (ISR'04).

13. A. Berenguer, J. Goncalves, S. Hosio, D. Ferreira, T. Anagnostopoulos, and V. Kostakos. (2017). Are Smartphones Ubiquitous?: An in-depth survey of smartphone adoption by seniors. IEEE Consumer Electronics Magazine, 6(1), 104-110.
14. Massimo Bergamasco, “The GLAD-IN-ART Project”; In Proceedings of Imagina ‘92 (Edited by Le Centre National de la Cinematographie (CNC); Monte Carlo, France; January, 1992; pp. II-7 to II-14.
15. Matteo Bianchi, Edoardo Battaglia, Mattia Poggiani, Simone Ciotti, and Antonio Bicchi. 2016. A Wearable Fabric-based display for haptic multi-cue delivery. In Proceedings of Haptics Symposium (HAPTICS 2016), 277-283. DOI: 10.1109/HAPTICS.2016.7463190
16. James Biggs and Mandayam A. Srinivasan. 2002. Tangential Versus Normal Displacements of Skin: Relative Effectiveness for Producing Tactile Sensations. In Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS 2002), 121-128. DOI: 10.1109/HAPTIC.2002.998949
17. Mark Billinghurst, and Thad Starner, (1999). Wearable devices: new ways to manage information. Computer, 32(1), 57-64.
18. K. Boven, M. Fejtl, A. Moeller, W. Nisch, and A. Stett. On Micro-Electrode Array Revival. In: Advances in Network Electrophysiology Using Multi-Electrode Arrays, 2006: 24-37.
19. V. Boxtel, Skin resistance during square-wave electrical pulses of 1 to 10 mA. Medical and Biological Engineering and Computing 15.6 (1977): 679-687.
20. Grigore Burdea. Haptic Issues in Virtual Environments. In Proceedings of Computer Graphics International (GCI ’00), 2000. https://doi.org/10.1109/CGI.2000.852345
21. Rehastim by Hasomed, https://www.hasomed.de/en/products.html, last accessed on 2018/02/02.
22. Baptiste Caramiaux, Marco Donnarumma, and Atau Tanaka. 2015. Understanding Gesture Expressivity through Muscle Sensing. ACM Trans. Comput.-Hum. Interact. 21, 6, Article 31 (January 2015), 26 pages. DOI: https://doi.org/10.1145/2687922
23. J. Chae, F. Bethoux, T. Bohinc, L. Dobos, T. Davis, A. Friedl. Neuromuscular stimulation for upper extremity motor and functional recovery in acute hemiplegia. Stroke. 1998;29(5): 975-9.
24. Liwei Chan, Chi-Hao Hsieh, Yi-Ling Chen, Shuo Yang, Da-Yuan Huang, Rong-Hao Liang, and Bing-Yu Chen. 2015. Cyclops: Wearable and Single-Piece Full-Body Gesture Input Devices. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 3001-3009. DOI: https://doi.org/10.1145/2702123.2702464
25. E. Chao, K. An, W. Cooney, and R. Linscheid. Biomechanics of the Hand. World Scientific Publishing, 1989.
26. Daniel Chen, Iain Anderson, Cameron Walker and Thor Besier. 2016. Lower Extremity Lateral Skin Stretch Perception for Haptic Feedback. Haptics IEEE Transactions, 9, 62-68, 2016. DOI: 10.1109/TOH.2016.2516012
27. A. Coleman (1969). Effect of unilateral isometric and isotonic contractions on the strength of the contralateral limb. Research quarterly. 40. 490-5. 10.1080/10671188.1969.10614867.

28. Heather Culbertson, Juan Jose Lopez Delgado, Katherine J. Kuchenbecker. One hundred data-driven haptic texture models and open-source methods for rendering on 3d objects. In Proc. IEEE Haptics Symposium, 2014. https://doi.org/10.1109/HAPTICS.2014.6775475
29. S. Daniel, S. Gallagher, (Eds.), Handbook of Phenomenology and Cognitive Science, Springer; 2010 edition, ISBN-10: 9048126452
30. T. Djajadiningrat, B. Matthews, and M. Stienstra. Easy doesn't do it: skill and expression in tangible aesthetics. Personal Ubiquitous Comput. 11, 8, 2007, 657-676.
31. Ayaka Ebisu, Satoshi Hashizume, Kenta Suzuki, Akira Ishii, Mose Sakashita, and Yoichi Ochiai. 2016. Stimulated percussions: techniques for controlling human as percussive musical instrument by using electrical muscle stimulation. In SIGGRAPH ASIA 2016 Posters (SA '16), Article 37.
32. Arthur Elsenaar and Remko Scha, Electric Body Manipulation as Performance Art: A Historical Perspective, in Leonardo Music Journal, December 2002, Vol. 12, Pages: 17-28.
33. Douglas Engelbart and William English. 1968. A research center for augmenting human intellect. In Proceedings of the December 9-11, 1968, fall joint computer conference, part I (AFIPS '68 (Fall, part I)). ACM, New York, NY, USA, 395-410. DOI=http://dx.doi.org/10.1145/1476589.1476645
34. Farzam Farbiz, Zhou Hao Yu, Corey Manders, and Waqas Ahmad. An electrical muscle stimulation haptic feedback for mixed reality tennis game. In ACM SIGGRAPH 2007 posters (SIGGRAPH '07), Article 140. https://doi.org/10.1145/1280720.1280873
35. Steven Feiner. 2002. Augmented reality: A new way of seeing. Scientific American, 286(4), 2002, 34–41.
36. Torsten Felzer and Rainer Nordmann. 2008. Using intentional muscle contractions as input signals for various hands-free control applications. In Proceedings of the 2nd International Convention on Rehabilitation Engineering & Assistive Technology (iCREATe '08), 87-91.
37. James Foley. Interfaces for advanced computing. Sci. Am. 257, 4, 126-135, 1987. http://dx.doi.org/10.1038/scientificamerican1087-126
38. Ryohei Fushimi, Eisuke Fujinawa, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. 2017. Vibrat-o-matic: producing vocal vibrato using EMS. In Proceedings of the 8th Augmented Human International Conference (AH '17). ACM, New York, NY, USA, Article 24, 5 pages. DOI: https://doi.org/10.1145/3041164.3041193
39. S. Gandevia, J. Smith, M. Crawford, U., Proske, and J. Taylor, Motor commands contribute to human position sense. Journal of Physiology, 2006, 571(3) 703-710.
40. E. Gardner and J. Martin. Coding of Sensory Information, Principles of Neural Science. McGraw-Hill, fourth edition, 2000.
41. Michael Gazzaniga, and G.R. Mangun (editors) Cognitive Neuroscience V. MIT Press.
42. James Gibson, The Ecological Approach to Visual Perception, Psychology Press, 1979, Chapter 8.

43. Ray Goertz, ANL Model E1, Master-Slave Manipulator, 1948, U.S. Atomic Energy Commission, Patent: US2632574 A, Granted in Mar 24, 1953.
44. Takashi Goto, Benjamin Tag, Kai Kunze, and Tilman Dingler. 2018. Towards Enhancing Emotional Responses to Media using Auto-Calibrating Electric Muscle Stimulation (EMS). In Proceedings of the 9th Augmented Human International Conference (AH '18). ACM, New York, NY, USA, Article 29, 2 pages. DOI: https://doi.org/10.1145/3174910.3174939
45. Erik Grönvall, Jonas Fritsch, and Anna Vallgårda. 2016. FeltRadio: Sensing and Making Sense of Wireless Traffic. In Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16). ACM, New York, NY, USA, 829-840. DOI: https://doi.org/10.1145/2901790.2901818
46. P. Gruenbaum, W. McNeely, H. Sowizral, T. Overman, and B. Knutson, Implementation of dynamic robotic graphics for a virtual control panel, Proc. Presence'97, 118–126.
47. Xiaochi Gu, Yifei Zhang, Weize Sun, Yuanzhe Bian, Dao Zhou, and Per Ola Kristensson. Dexmo: An Inexpensive and Lightweight Mechanical Exoskeleton for Motion Capture and Force Feedback in VR. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), 1991-1995. https://doi.org/10.1145/2858036.2858487
48. Sean Gustafson, Daniel Bierwirth, and Patrick Baudisch. 2010. Imaginary interfaces: spatial interaction with empty hands and without visual feedback. In Proceedings of the 23nd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 3-12. DOI: https://doi.org/10.1145/1866029.1866033
49. Sean Gustafson, Bernhard Rabe, and Patrick Baudisch. 2013. Understanding palm-based imaginary interfaces: the role of visual and tactile cues when browsing. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 889-898. DOI: https://doi.org/10.1145/2470654.2466114
50. Sean Gustafson, Christian Holz, and Patrick Baudisch. 2011. Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11). ACM, New York, NY, USA, 283-292. DOI: https://doi.org/10.1145/2047196.2047233
51. Shin Hanagata and Yasuaki Kakehi. 2018. Paralogue: A Remote Conversation System Using a Hand Avatar which Postures are Controlled with Electrical Muscle Stimulation. In Proceedings of the 9th Augmented Human International Conference (AH '18). ACM, New York, NY, USA, Article 35, 3 pages. DOI: https://doi.org/10.1145/3174910.3174951
52. Red Hands, http://en.wikipedia.org/wiki/Red_hands, last accessed on 2018/06/06.
53. Blake Hannaford and Allison M. Okamura, Haptics (chapter), Handbook of Robotics, 2008, Oussama Khat-ib, Bruno Siciliano(eds.), Springer, pp 719-739.
54. Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: appropriating the body as an input surface. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10). ACM, New York, NY, USA, 453-462. DOI: https://doi.org/10.1145/1753326.1753394

55. Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011. OmniTouch: wearable multitouch interaction everywhere. In Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11). ACM, New York, NY, USA, 441-450. DOI: https://doi.org/10.1145/2047196.2047255
56. Chris Harrison, Shilpa Ramamurthy, and Scott E. Hudson. 2012. On-body interaction: armed and dangerous. In Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12), Stephen N. Spencer (Ed.). ACM, New York, NY, USA, 69-76. DOI: https://doi.org/10.1145/2148131.2148148
57. Shoichi Hasegawa, Ishikawa Toshiaki, Naoki Hashimoto, Marc Salvati, Hironori Mitake, Yasuharu Koike, and Makoto Sato. 2006. Human-scale haptic interaction with a reactive virtual human in a real-time physics simulator. Comput. Entertain. 4, 3, Article 9 (July 2006). DOI=http://dx.doi.org/10.1145/1146816.1146831
58. Mahmoud Hassan, Florian Daiber, Frederik Wiehr, Felix Kosmalla, and Antonio Krüger. 2017. FootStriker: An EMS-based Foot Strike Assistant for Running. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1, 1, Article 2 (March 2017), 18 pages. DOI: https://doi.org/10.1145/3053332
59. Mariam Hassib, Max Pfeiffer, Stefan Schneegass, Michael Rohs, and Florian Alt. 2017. Emotion Actuator: Embodied Emotional Feedback through Electroencephalography and Electrical Muscle Stimulation. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 6133-6146. DOI: https://doi.org/10.1145/3025453.3025953
60. August Heidland, Gholamreza Fazeli, André Klassen, Katarina Sebekova, Hans Hennemann, Udo Bahner, and Biagio Di Iorio. 2013. Neuromuscular electrostimulation techniques: historical aspects and current possibilities in treatment of pain and muscle waisting. Clinical Nephrology 79 Suppl 1: S12-23.
61. Steven Henderson and Steven Feiner. 2009. Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret. In Proceedings of the 2009 8th IEEE International Symposium on Mixed and Augmented Reality (ISMAR '09), 135-144. DOI: 10.1109/ISMAR.2009.5336486
62. Anuruddha Hettiarachchi and Daniel Wigdor. 2016. Annexing Reality: Enabling Opportunistic Use of Everyday Objects as Tangible Proxies in Augmented Reality. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 1957-1967. DOI: https://doi.org/10.1145/2858036.2858134
63. Ken Hinckley, Patrick Baudisch, Gonzalo Ramos, and Francois Guimbretiere. 2005. Design and analysis of delimiters for selection-action pen gesture phrases in scriboli. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '05). ACM, New York, NY, USA, 451-460. DOI=http://dx.doi.org/10.1145/1054972.1055035
64. John Hollerbach and Stephen Jacobsen. Haptic Interfaces for Teleoperation and Virtual Environments. In Proc. Workshop on Simulation and Interaction in Virtual Environments'95, 13–15. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.52.978
65. Microsoft HoloLens, https://www.microsoft.com/microsoft-hololens/en-us/developers, last accessed on 2017/09/09.

66. Christian Holz, Tovi Grossman, George Fitzmaurice, and Anne Agur. 2012. Implanted user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12). ACM, New York, NY, USA, 503-512. DOI: http://dx.doi.org/10.1145/2207676.2207745
67. Jonathan Hook, Thomas Nappey, Steve Hodges, Peter Wright, and Patrick Olivier. 2014. Making 3D printed objects interactive using wireless accelerometers. In CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14). ACM, New York, NY, USA, 1435-1440. DOI: https://doi.org/10.1145/2559206.2581137
68. T. Hoshi, M. Takashami, T. Iwamoto, and H. Shinoda, Noncontact tactile display based on radiation pressure of Airborne Ultrasound. In IEEE Trans. Haptics. 3, 2010, 155–165.
69. GearVR, http://samsung.com/global/microsite/gearvr/, last accessed on 11/04/2015.
70. HASOMED, http://www.hasomed.de, last accessed in 2018/05/16.
71. Vuforia, https://developer.vuforia.com, last accessed on 2018/02/02.
72. Hiroshi Ishii and Brygg Ullmer. 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In Proceedings of the ACM SIGCHI Conference on Human factors in computing systems (CHI '97). ACM, New York, NY, USA, 234-241. DOI=http://dx.doi.org/10.1145/258549.258715
73. K. Ito, S. Takahiro, and K. Toshiyuki. Lower-limb Joint Torque and Position Controls by Functional Electrical Stimulation (FES). Complex Medical Engineering. Springer Japan, 2007. 239-249.
74. Patricia Ivette Cornelio Martinez, Silvana De Pirro, Chi Thanh Vi, and Sriram Subramanian. 2017. Agency in Mid-air Interfaces. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 2426-2439. DOI: https://doi.org/10.1145/3025453.3025457
75. Seungzoo Jeong, Naoki Hashimoto, and Sato Makoto. 2004. A novel interaction system with force feedback between real - and virtual human: an entertainment system: "virtual catch ball". In Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology (ACE '04). ACM, New York, NY, USA, 61-66. DOI=http://dx.doi.org/10.1145/1067343.1067350
76. M. Johnson (2007). Transcutaneous Electrical Nerve Stimulation: Mechanisms, Clinical Application and Evidence. Reviews in Pain, 1(1), 7–11. http://doi.org/10.1177/204946370700100103
77. Thorsten Karrer, Moritz Wittenhagen, Leonhard Lichtschlag, Florian Heller, and Jan Borchers. 2011. Pinstripe: eyes-free continuous input on interactive clothing. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 1313-1322. DOI: https://doi.org/10.1145/1978942.1979137
78. Idin Karuei, Karon E. MacLean, Zoltan Foley-Fisher, Russell MacKenzie, Sebastian Koch, and Mohamed El-Zohairy. 2011. Detecting vibrations across the body in mobile contexts. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 3267-3276. DOI: https://doi.org/10.1145/1978942.1979426

79. Manami Katoh, Narihiro Nishimura, Maki Yokoyama, Taku Hachisu, Michi Sato, Shogo Fukushima, and Hiroyuki Kajimoto. 2013. Optimal selection of electrodes for muscle electrical stimulation using twitching motion measurement. In Proceedings of the 4th Augmented Human International Conference (AH '13). ACM, New York, NY, USA, 237-238. DOI=http://dx.doi.org/10.1145/2459236.2459279
80. Hwan Kim, Minhwan Kim, and Woohun Lee. 2016. HapThimble: A Wearable Haptic Device towards Usable Virtual Touch Screen. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 3694-3705. DOI: https://doi.org/10.1145/2858036.2858196
81. David Kim, Otmar Hilliges, Shahram Izadi, Alex D. Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. 2012. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. In Proceedings of the 25th annual ACM symposium on User interface software and technology (UIST '12). ACM, New York, NY, USA, 167-176. DOI: https://doi.org/10.1145/2380116.2380139
82. J. Kim, W. Lee, …, J. Kim, J. (2016). Differences in strength-duration curves of electrical diagnosis by physiotherapists between DJ-1 homozygous knockout and wild-type mice: a randomized controlled pilot trial. Journal of Physical Therapy Science, 28(5), 1595–1598. http://doi.org/10.1589/jpts.28.1595
83. Majken Kirkegaard Rasmussen, Erik Grönvall, Sofie Kinch, and Marianne Graves Petersen. 2013. "It's alive, it's magic, it's in love with you": opportunities, challenges and open questions for actuated interfaces. In Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration (OzCHI '13), ACM, New York, NY, USA, 63-72. DOI: http://dx.doi.org/10.1145/2541016.2541033
84. Jarrod Knibbe, Paul Strohmeier, Sebastian Boring, and Kasper Hornbæk. 2017. Automatic Calibration of High Density Electric Muscle Stimulation. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1, 3, Article 68 (September 2017), 17 pages. DOI: https://doi.org/10.1145/3130933
85. Benjamin Knoerlein, Gábor Székely, and Matthias Harders. 2007. Visuo-haptic collaborative augmented reality ping-pong. In Proceedings of the international conference on Advances in computer entertainment technology (ACE '07). ACM, New York, NY, USA, 91-94. DOI=http://dx.doi.org/10.1145/1255047.1255065
86. James Kramer. Force feedback and Texture Simulating Interface device, U.S. Patent 5, 184, 319.
87. Alexander Kron, and Guenther Schmidt. Multi-fingered Tactile Feedback from Virtual and Remote Environments. In Proc. HAPTICS'03, 16–23. https://doi.org/10.1109/HAPTIC.2003.1191219
88. Ernst Kruijff, Dieter Schmalstieg, and Steffi Beckhaus. 2006. Using neuromuscular electrical stimulation for pseudo-haptic feedback. In Proceedings of the ACM symposium on Virtual reality software and technology (VRST '06). ACM, New York, NY, USA, 316-319. http://dx.doi.org/10.1145/1180495.1180558
89. Shinobu Kuroki, Hiroyuki Kajimoto, Hideaki Nii, Naoki Kawakami, and Susumu Tachi. (2007). Proposal for tactile sense presentation that combines electrical and mechanical stimulus. 121 - 126. http://doi.org/10.1109/WHC.2007.92.

90. Anthony Kurtz. Tactile feel apparatus for use with robotic operations, U.S. Patent 650, 643, filed January 8, 2007, issued July 10, 2008.
91. Bruno Latour. Where are the missing masses? The sociology of a few mundane artifacts. In Shaping technology/building society, 1992, 225-58.
92. Thin lens simulation with Python. http://www.frantzmartinache.com/blog/?p=84, last accessed 2017/09/09.
93. P. Letier, M. Avraam, Samuel Veillerette, Mihaita Horodinca, M. de Bartolomei, A. Schiele, and A. Preumont (2008). SAM : A 7-DOF portable arm exoskeleton with local joint control. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS. 3501 - 3506. 10.1109/IROS.2008.4650889.
94. Kevin Li, Designing easily learnable eyes-free interaction, Ph.D. Thesis. UC San Diego, 2009.
95. Cheng-Yuan Li, Yen-Chang Chen, Wei-Ju Chen, Polly Huang, and Hao-hua Chu. 2013. Sensor-embedded teeth for oral activity recognition. In Proceedings of the 2013 International Symposium on Wearable Computers (ISWC '13). ACM, New York, NY, USA, 41-44. DOI: https://doi.org/10.1145/2493988.2494352
96. Chunyuan Liao, François Guimbretière, and Ken Hinckley. 2005. PapierCraft: a command system for interactive paper. In Proceedings of the 18th annual ACM symposium on User interface software and technology (UIST '05). ACM, New York, NY, USA, 241-244. DOI=http://dx.doi.org/10.1145/1095034.1095074
97. Yi-Chi Liao, Shun-Yao Yang, Rong-Hao Liang, Liwei Chan, and Bing-Yu Chen. 2015. ThirdHand: wearing a robotic arm to experience rich force feedback. In SIGGRAPH Asia 2015 Emerging Technologies (SA '15). ACM, New York, NY, USA, Article 24, 1 pages. DOI: https://doi.org/10.1145/2818466.2818487
98. Joshua Lifton and Joseph A. Paradiso, Dual Reality: Merging the Real and Virtual, International Conference on Facets of Virtual Environments FaVE 2009: pp. 12-28. DOI:10.1007/978-3-642-11743-5_2
99. Robert Lindeman, Yasuyuki Yanagida, Haruo Noma, and Kenichi Hosaka. Wearable vibrotactile systems for virtual contact and information display. In Virtual Reality 9, 2, 203–213, 2006. http://dx.doi.org/10.1007/s10055-005-0010-6
100. Therese Lloyd, Giovanni de Domenico, Geoffrey Strauss, and Kevin Singer, A Review of the Use of Electro-Motor Stimulation in Human Muscles, Australian Journal of Physiotherapy, Volume 32, Issue 1, 1986, Pages 18-30, ISSN 0004-9514, https://doi.org/10.1016/S0004-9514(14)60640-1.
101. Daito Manabe, Face Visualizer, 2008, http://www.daito.ws/en/work/smiles.html, last accessed in 2018/02/01.
102. Paul Marshall. (2007, February). Do tangible interfaces enhance learning?. In Proceedings of the 1st international conference on Tangible and embedded interaction (pp. 163-170). DOI=http://dx.doi.org/10.1145/1226969.1227004
103. William McNeely. Robotic graphics: a new approach to force feedback for virtual reality. In Proceedings of the Annual IEEE International Symposium on Virtual Reality, 336–341, 1999. https://doi.org/10.1109/VRAIS.1993.380761
104. Paul Milgram and Fumio Kishino. 1994. A Taxonomy of mixed reality Visual Displays. IEICE Transactions on Information and Systems E77-D,1321–1329.

105. Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino. 1995. Augmented reality: A class of displays on the reality-virtuality continuum. In Proceedings of SPIE 2351, Telemanipulator and Telepresence Technologies, 282-292.

106. Neil Mizen, Powered exoskeletal apparatus for amplifying human strength in response to normal body movements, Patent US3449769 A, Granted 17 Jun 1969.

107. Jun Murayama, Laroussi Bougrila, Yanlinluo Katsuhito Akahane, Shoichi Hasegawa, Béat Hirsbrunner and Makoto Sato. SPIDAR G&G: a two-handed haptic interface for bimanual VR interaction. In Proc. EuroHaptics'04, 138–146.

108. Kazuki Nagai, Soma Tanoue, Katsuhito Akahane, and Makoto Sato. Wearable 6-DoF wrist haptic device "SPIDAR-W". In SIGGRAPH Asia 2015 Haptic Media And Contents Design (SA '15), Article 19. http://dx.doi.org/10.1145/2818384.2818403

109. Tao Ni and Patrick Baudisch. 2009. Disappearing mobile devices. In Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST '09). ACM, New York, NY, USA, 101-110. DOI: https://doi.org/10.1145/1622176.1622197

110. Arinobu Niijima, Takashi Isezaki, Ryosuke Aoki, Tomoki Watanabe, and Tomohiro Yamada. 2018. Controlling Maximal Voluntary Contraction of the Upper Limb Muscles by Facial Electrical Stimulation. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18). ACM, New York, NY, USA, Paper 394, 7 pages. DOI: https://doi.org/10.1145/3173574.3173968

111. Arinobu Niijima, and Takefumi Ogawa. (2016). A proposal of virtual food texture by electric muscle stimulation. 1-6. 10.1109/ICMEW.2016.7574698.

112. Jun Nishida and Kenji Suzuki. 2017. bioSync: A Paired Wearable Device for Blending Kinesthetic Experience. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17). ACM, New York, NY, USA, 3316-3327. DOI: https://doi.org/10.1145/3025453.3025829

113. Geoff Norman. 2010. Likert scales, levels of measurement, and the “laws” of statistics. In Advances in Health Sciences Education 15, 5, 625-632, 2010. https://dx.doi.org/10.1007/s10459-010-9222-y

114. Donald Norman. The Design of Everyday Things: Revised and Expanded Edition. Basic Books, 2013.

115. Tesseract OCR. http://github.com/tesseract-ocr, last accessed 2018/05/06.

116. Ohan Oda and Steven Feiner. 2010. Rolling and shooting: two augmented reality games. In CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10). ACM, New York, NY, USA, 3041-3044. DOI: https://doi.org/10.1145/1753846.1753914

117. Ohan Oda, Carmine Elvezio, Mengu Sukan, Steven Feiner, and Barbara Tversky. 2015. Virtual Replicas for Remote Assistance in Virtual and Augmented Reality. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15). ACM, New York, NY, USA, 405-415. DOI: https://doi.org/10.1145/2807442.2807497

118. Krista Overvlieta, Jeroen Smeetsa, and E. Brennera. The use of proprioception and tactile information in haptic search. In Acta Psychologica 129, 1, 83–90, 2008. https://doi.org/10.1016/j.actpsy.2008.04.011

119. Jerome Pasquero, Scott J. Stobbe, and Noel Stonehouse. 2011. A haptic wristwatch for eyes-free interactions. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 3257-3266. DOI: https://doi.org/10.1145/1978942.1979425

120. Mark Paterson, The Senses of Touch: Haptics, Affects and Technologies, Bloomsbury Academic, 978-1845204792

121. Hunter Peckham, Jayme S. Knutson (2005). Functional Electrical Stimulation for Neuromuscular Applications, Annual Review of Biomedical Engineering, 2015, 7:1, 327-360.

122. Max Pfeiffer, Stefan Schneegass, Florian Alt, and Michael Rohs. 2014. Let me grab this: a comparison of EMS and vibration for haptic feedback in free-hand interaction. In Proceedings of the 5th Augmented Human International Conference (AH '14). ACM, New York, NY, USA, Article 48, 8 pages. DOI=http://dx.doi.org/10.1145/2582051.2582099

123. Max Pfeiffer, Tim Dünte, Stefan Schneegass, Florian Alt, and Michael Rohs. 2015. Cruise Control for Pedestrians: Controlling Walking Direction using Electrical Muscle Stimulation. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 2505-2514. http://doi.acm.org/10.1145/2702123.2702190

124. Max Pfeiffer, Tim Duente, and Michael Rohs. 2016. Let your body move: a prototyping toolkit for wearable force feedback with electrical muscle stimulation. In Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '16). ACM, New York, NY, USA, 418-427. DOI: http://dx.doi.org/10.1145/2935334.2935348

125. Senseable Phantom Haptic Arm. http://www.sensable.com, last accessed on 10/09/2012.

126. Claudio Pinhanez. 2001. The Everywhere Displays Projector: A Device to Create Ubiquitous Graphical Interfaces. In Proceedings of the 3rd international conference on Ubiquitous Computing (UbiComp '01), 315-331.

127. Henning Pohl, Kasper Hornbæk, and Jarrod Knibbe. 2018. Wanding Through Space: Interactive Calibration for Electric Muscle Stimulation. In Proceedings of the 9th Augmented Human International Conference (AH '18). ACM, New York, NY, USA, Article 19, 5 pages. DOI: https://doi.org/10.1145/3174910.3174948

128. U. Proske, and S. Gandevia, The Proprioceptive Senses: Their Roles in Signaling Body Shape, Body Position and Movement, and Muscle Force, In Physiological Reviews 2012, Vol. 92 no. 4,1651-1697

129. Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. RetroFab: A Design Tool for Retrofitting Physical Interfaces using Actuators, Sensors and 3D Printing. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 409-419. DOI: https://doi.org/10.1145/2858036.2858485

130. M. Rank, Z. Shi, H. Müller, and S. Hirche, (2010) Perception of Delay in Haptic Telepresence Systems, In Presence, 19(5), 389–399.

131. J. Rosen, J. C. Perry, N. Manning, S. Burns, B. Hannaford, The Human Arm Kinematics and Dynamics During Daily Activities – Toward a 7 DOF Upper Limb Powered Exoskeleton, - ICAR 2005 – Seattle WA, July 2005.

132. Anne Roudaut, Andreas Rau, Christoph Sterz, Max Plauth, Pedro Lopes, and Patrick Baudisch. 2013. Gesture output: eyes-free output using a force feedback touch surface. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 2547-2556. DOI: https://doi.org/10.1145/2470654.2481352

133. David Ruis, Russell Polhemus, Wayne Book, Robotic exercise machine and method, Patent US4235437 A, Granted 25 Nov 1980.

134. Scott Saponas, Desney Tan, Dan Morris, Ravin Balakrishnan, Jim Turner, and James A. Landay. 2009. Enabling always-available input with muscle-computer interfaces. In Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST '09). ACM, New York, NY, USA, 167-176. DOI: https://doi.org/10.1145/1622176.1622208

135. Nitin Sawhney, and Chris Schmandt. (1998). Speaking and Listening on the Run: Design for Wearable Audio Computing. 108 - 115. 10.1109/ISWC.1998.729536.

136. T. Shannon, Tactile communication attachment, Patent US3780225 A, Granted Dec 18, 1973.19

137. William Sherman and Alan Craig. Understanding Virtual Reality: Interface, Application, and Design. ed. Morgan Kaufmann, Series in Computer Graphics.

138. Timothy Simon, Ross Smith, and Bruce Thomas. Wearable jamming mitten for virtual environment haptics. In Proceedings of the 2014 ACM International Symposium on Wearable Computers (ISWC '14), 67-70. http://dx.doi.org/10.1145/2634317.2634342

139. Lattice-Boltzmann Simulation of a 2D Fluid. http://physics.weber.edu/schroeder/fluids, last accessed 2017/09/09.

140. A. Skulmowski, S. Pradel, T. Kühnert, G. Brunnett, and G. Rey, (2016). Embodied learning using a tangible user interface: the effects of haptic perception and selective pointing on a spatial learning task. Computers & Education, 92, 64-75.

141. Rajinder Sodhi, Ivan Poupyrev, Matthew Glisson, and Ali Israr. 2013. AIREAL: interactive tactile experiences in free air. ACM Trans. Graph. 32, 4, Article 134 (July 2013), 10 pages. DOI: https://doi.org/10.1145/2461912.2462007

142. Open Source OCR packages. http://www.mathstat.dal.ca/~selinger/ocr-test, last accessed in 2017/09/09.

143. Jurgen Steimle. (2016). Skin--The Next User Interface. Computer. 49. 83-87. 10.1109/MC.2016.93.

144. Maria Stokes, Max Blythe (2001). Muscle Sounds in physiology, sports science and clinical investigation. Oxford: Medintel. ISBN 0-9540572-0-1.

145. P. Strojnik, A. Kralj, and I. Ursic. Programmed six-channel electrical stimulator for complex stimulation of leg muscles during walking. In IEEE Trans. Biomed. Eng. 26, 112. https://doi.org/10.1109/TBME.1979.326520

146. Hong Z. Tan and Alex Pentland. 2005. Tactual displays for sensory substitution and wearable computers. In ACM SIGGRAPH 2005 Courses (SIGGRAPH '05), John Fujii (Ed.). ACM, New York, NY, USA, Article 105 . DOI: https://doi.org/10.1145/1198555.1198611

147. Emi Tamaki, Takashi Miyaki, and Jun Rekimoto. 2011. PossessedHand: techniques for controlling human hands using electrical muscles stimuli. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 543-552. http://dx.doi.org/10.1145/1978942.1979018
148. M. Tarata (2003). Mechanomyography versus Electromyography, in monitoring the muscular fatigue. BioMedical Engineering OnLine, 2, 3. http://doi.org/10.1186/1475-925X-2-3
149. Sho Tatsuno, Tomohiko Hayakawa, and Masatoshi Ishikawa: Supportive Training System for Sports Skill Acquisition Based on Electrical Stimulation, 2017 IEEE World Haptics Conference (WHC) (Munich, 2017.6.8) /Proceedings of the 2017 IEEE World Haptics Conference (WHC), pp.466-471, DOI: 10.1109/SMC.2016.7844413
150. Anoto Technology, http://www.anoto.com, last accessed in 2017/01/02
151. Jonas Togler, Fabian Hemmert, and Reto Wettach. 2009. Living interfaces: the thrifty faucet. In Proceedings of the 3rd International Conference on Tangible and Embedded Interaction (TEI '09). ACM, New York, NY, USA, 43-44. DOI: https://doi.org/10.1145/1517664.1517680
152. Dzmitry Tsetserukou, Katsunari Sato, and Susumu Tachi. ExoInterfaces: novel exosceleton haptic interfaces for virtual reality, augmented sport and rehabilitation. In Proceedings of the 1st Augmented Human International Conference (AH '10), article 1. http://dx.doi.org/10.1145/1785455.1785456
153. Peter-Paul Verbeek. Materializing Morality. Design Ethics and Technological Mediation, In Science, Technology and Human Values 31, 2006, (3):361-380.
154. L. Walsh, J. Smith, S. Gandevia, and J. Taylor, The combined effect of muscle contraction history and motor commands on human position sense. Journal of Experimental Brain Research, 2009; 195(4): 603-10.
155. Saiwen Wang, Jie Song, Jaime Lien, Ivan Poupyrev, and Otmar Hilliges. 2016. Interacting with Soli: Exploring Fine-Grained Dynamic Gesture Recognition in the Radio-Frequency Spectrum. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16). ACM, New York, NY, USA, 851-860. DOI: https://doi.org/10.1145/2984511.2984565
156. Sean White and Steven Feiner. 2009. SiteLens: situated visualization techniques for urban site visits. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09). ACM, New York, NY, USA, 1117-1120. DOI: https://doi.org/10.1145/1518701.1518871
157. E. Whittaker (1951), A history of the theories of aether and electricity. Vol 1, DOI: https://archive.org/details/historyoftheorie00whitrich
158. Yasuyoshi Yokokohji, Ralph L. Hollis, and Takeo Kanade. 1996. What you can see is what you can feel-development of a visual/haptic interface to virtual environment." In Proceedings of the IEEE 1996 Virtual Reality Annual International Symposium, 46-53.