Bidirectional Force Input: Increasing and Decreasing Values on Mobile Devices with the Thumb

Master's Thesis
submitted to the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Andreas Link

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Univ.-Prof. Dr.-Ing. Frank Flemisch

Registration date: 01/09/2017
Submission date: 29/09/2017

Eidesstattliche Versicherung

_____________________________ _____________________________

Name, Vorname Matrikelnummer

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

____

____

____

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

_____________________________ _____________________________

Ort, Datum Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

_____________________________ _____________________________

Ort, Datum Unterschrift

Contents

List of Figures

List of Tables

Abstract

Force depicts an *additional input channel* and allows to overcome *usability issues* when operating *mobile devices*, like *smartphones*, *single-handed* and only using the *thumb*. In this manner, the *user's thumb* can stay within its *comfortable interaction range*, and uses *force input* to perform *value manipulations* from a *static location*. Nevertheless, *force* is limited in the way that it is *unidirectional*, and hence does not allow for *bidirectional input*. To overcome this *limitation*, this *thesis* follows a *systematic procedure* to find an *appropriate solution* to the *bidirectional problem*.

In this thesis *eighteen designs* are proposed that enable *bidirectional force input* from a *static location*, and are built from *three essential components*, namely *pressure mapping*, *direction-* and *pressure-control mechanism*. In this regard, considerable care has been taken to consider *people's force-sensing capabilities* as well as the *ergonomic characteristics* of the *human thumb*. Within the *first study*, *thumb roll*, *quick pulse* and *natural mapping*, i.e., *combinations* of the *three essential components*, could be identified to yield *best results* within the *first investigation*. Finally, the *second study* focussed on *remaining designs*, and *compared* their *performance* against a *baseline condition*.

Results revealed that *thumb roll* and *quick pulse* are *appropriate solutions* to the *bidirectional problem*, since they enable *bidirectional force input* with *great accuracy* of almost 99%. Even though *thumb roll* performed 910ms slower than *baseline*, we are confident that *users* can become *faster* with *further training*. Taken together, we believe that *findings* are especially beneficial to *interaction designers*, since they provide a *first solution* to the *bidirectional problem*, and hence make *force input* applicable to a *variety* of *application domains*.

Überblick

Druck stellt einen zusätzlichen Eingabekanal zur Verfügung, der bspw. dazu verwendet werden kann, einhändiges Bedienen größerer *Smartphones* zu erleichtern. So kann der Daumen des Nutzers weiterhin in seinem natürlichen Interaktionsbereich im unteren Teil des Smartphones verweilen, von wo aus zusätzliche Funktionalität durch verschiedene Druckstufen gesteuert werden kann. Das *Problem* besteht jedoch darin, dass Druck ein *einseitig gerichteter Parameter* ist, weshalb Wertmanipulationen nur in eine Richtung möglich sind. Um dieser Limitierung entgegenzuwirken, verfolgt diese Arbeit ein *systematisches Verfahren* mit dem Ziel, eine geeignete Lösung für das "Bidirektionale Problem" zu finden.

Im Rahmen dieser Arbeit wurden *achtzehn* Interaktionstechniken entwickelt, die *bidirektionale Wertmanipulationen* mittels *Druckinteraktion* ermöglichen, und sich aus drei grundlegenden Komponenten, d.h. aus der Druckabbildung, dem Richtungs- und Druckkontrollmechanismus, zusammensetzen. Hierbei wurde insbesondere das *Druckempfinden* und die *ergonomischen Eigenschaften* des *menschlichen Daumens* berücksichtigt. Innerhalb einer *ersten Studie* konnten drei Kombinationen, d.h. die *Daumenrolle*, der *schnelle Impuls* und die *natürliche Zuordnung*, aus den zuvor genannten Komponenten identifiziert werden, die innerhalb eines *ersten Experiments* die besten Ergebnisse erzielen konnten. Diese wurden schließlich in einer *zweiten Studie* anhand einer Vergleichskondition gegenübergestellt.

Unsere Ergebnisse zeigen, dass sowohl die *Daumenrolle* als auch der *schnelle Impuls* geeignete Lösungen für das *bidireaktionale Problem* darstellen, da sie von Nutzern mit hoher Präzison von bis zu 99% angewendet werden können. Obwohl die *Daumenrolle* bis zu 910ms langsamer als die Vergleichskondition war, sind wir zuversichtlich, dass sich diese Unterschiede mit zusätzlichen Training deutlich minimieren lassen. Wir glauben, dass Erkenntnisse dieser Arbeit insbesondere für Interaktionsdesigner relevant sind, da sie eine mögliche Lösung für das *bidirektionale Problem* liefern, um Druckinteraktion einer Vielzahl weiterer Anwendungsgebiete zugänglich zu machen.

Acknowledgements

I would like to acknowledge the following people, without whose help this work would not have been possible:

First, I would like to thank my supervisor Christian Corsten for the *great support* I got throughout my *thesis*. The *initial schedule* as well as our *weekly meetings* supported me to keep track, and get *helpful advice* whenever needed. In addition, I really appreciate the *feedback* I got for my *proposal talk*, since it helped me a lot to get my *main points straight* in a minimum of time.

I am also particularly grateful to my thesis advisor Prof. Dr. Jan Borchers, who provided me with a *working environment* that offered everything I needed to conduct my *research*, and gave *valuable suggestions* for *bidirectional interaction designs*, as proposed in this *thesis*.

I also gratefully acknowledge Univ.-Prof. Dr.-Ing. Frank Flemisch, my second examiner, and Yigiterkut Canpolat from the chair and institute of industrial engineering and ergonomics, who offered *valuable insights* into the field of *thumb ergonomics*.

In addition, I wish to thank Dr. Simon Völker for his comments and suggestions regarding *implementation details* of *bidirectional designs*, as proposed in this *thesis*.

Thanks are also due to all *participants*, who took part in my *studies* and provided me with *constructive feedback* for each of the *bidirectional designs*.

Finally, I especially want to thank my *family* and *friends*, for the *great support* I got, and for motivating me to look for *solutions* that initially seemed to be unsolvable.

Andreas

Conventions

Throughout this thesis we use the following conventions.

Text conventions

For the purpose of *politeness*, the *thesis* is written in *first person plural form*.

Marginalia are included to summarize *important aspects* alongside the *thesis*.

This is an *important aspect*.

The thesis is written in American English.

Chapter 1

Introduction

Everyday tasks like *grabbing*, *moving* or *holding* an *object* strongly depend on *people's ability* to *control* and *exert force* with *great accuracy* and *precision*. As an example, placing an *object* at *different locations*, requires *load-forces* that lift the *object* from its *underlying surface*, as well as *grip-forces* that *keep* the *object* in a *stable position* [Johansson and Flanagan, 2009]. Hence, *people* exhibit *profound pressure-control capabilities* and are *well-familiar* with *pressure-based interaction*.

Humans exhibit profound *pressure-control capabilities*.

Recent work in the *field* of *human computer interaction* has taken advantage of *force input* to improve the *expressiveness* of *conventional input modalities*, like *mouse/keyboard-, tablet-,* or *pen-based interaction*, by distinguishing *multiple states*, functionality is mapped to [Cechanowicz et al., 2007, de Jong et al., 2010, Buxton et al., 1985, Ramos et al., 2007]. Equally important, *force* does not require *significant changes* in *hand posture*, and can be controlled from a *static location*. These *characteristics* are *especially beneficial* to overcome *limitations* within *one-handed use*, like *reachability-* or *occlusion issues*, that people *experience* when operating *mobile devices*, like *smartphones*, *single-handed* and only using their *thumb*.

Force can overcome limitations of *one-handed use*.

However, *force* is a *one-way continuous parameter*, and hence is *not suited* for many *application domains* [Mandalapu and Subramanian, 2011]. That's why this *thesis* follows a *systematic procedure* to *alleviate* this *issue*, and enable *bidirectional value manipulations* through *force-input* from a *static location*.

Limitation: Force input is *unidirectional*.

Two studies evaluated *proposed bidirectional designs* in terms of *user preference* and *performance*.

In this *thesis*, we looked at the *movement capabilities* of the *human thumb* and explored the *design space* of *bidirectional force input*, from which *eighteen designs* were derived. To evaluate their *appropriateness* regarding *user preference* and *performance*, *two studies* were performed.

Thumb roll and *quick pulse* enable *bidirectional force input* with *high accuracy* of $\approx 99\%$.

While the *first study* identified which *combination* of the *three required components* for *bidirectional force input*, i.e., *pressure-mapping*, *direction-* as well as *pressure-control mechanism*, *performs best* and is *most preferred*, the *second study* focussed on *remaining designs*, namely *quick pulse*, *thumb roll* and *natural mapping*, by *evaluating* their *performance* against a *baseline condition*. Results revealed that *thumb roll* and *quick pulse* are *appropriate solutions* for *bidirectional force input*, since they achieved *high accuracy* of almost 99%. Although *thumb roll* was $\approx 910ms$ slower compared to a *baseline-condition*, we are confident that *people* become *faster* with *further practice*.

Consequently this *thesis* is *structured* as follows:

- *Chapter 2* provides an *overview* of how *force input* can overcome *reachability-* and *occlusion issues* within *one-handed use*, and refers to the *anatomy* of the *human thumb* as well as *resulting limitations*. Moreover, *force characteristics* along with the *ingredients* for *pressure-based interaction* are discussed. Finally, the *chapter* concludes with the *bidirectional problem* and refers to a *systematic procedure*, derived from our *research questions*.
- To *overcome* the *bidirectional problem*, *Chapter 3* provides the *necessary background knowledge* and refers to *related work* in the *area* of *thumb ergonomics* and *pressure-based interaction modalities*. In this manner, the *thumb's movement capabilities* along with *several examples* of *force input* in context of *multitouch/tablet-*, *mouse/keyboard-*, *pen-based-* or *mobile device interaction* are discussed. In addition, *further directions* are stated.
- Based on *knowledge* from the *previous chapter*, *Chapter 4* addresses the *first research question* and proposes *several bidirectional designs*, including their *functional concepts* and *intended use-case*. In addition, *three essential components* for *bidirectional force input* are discussed. Finally, the *chapter* concludes with *implementation details* regarding the utilized *apparatus* and *architecture*.

- Having *addressed* the *first research question*, *Chapter 5* contains an *empirical evaluation* of *proposed bidirectional designs*, and *examines* which *combination* of *pressure-mapping*, *direction-* and *pressure-control mechanism performs best* and is *most preferred*. In this regard, the *chapter* refers to the *study design*, including *hypotheses*, the *study's task* as well as *important design decisions* made. In addition, *results* and *implications* are discussed.
- Finally, *Chapter 6* draws the *reader's attention* to the *second research question* and focusses on *remaining designs*, namely *quick pulse*, *thumb roll* and *natural mapping*, by comparing their performance against a *baseline condition*. Please be aware that the *chapter* only refers to *important changes* regarding the *study design*, since the *target-acquisition* and *selection-task* is similar to the *one* of the *previous study*. Finally, the *chapter* concludes with the *second study's results* as well as *important findings* that are summarized in *Chapter 7*.

Note that in the *literature*, *pressure (P)* is often equated with *force (F)*, even though the *meaning* of *both terms* is not the same. According to Giancoli [2005], *pressure* is defined as the amount of *force* per *unit area (A)* that acts *orthogonal* to the *underlying surface*. Consequently, *pressure* is defined by the following *equation*, yielding $[N/m^2]$ as *measurement unit*:

pressure $\neq$ force.

$$P = \frac{F}{A}$$

However, for the sake of *simplicity*, we use *both terms*, i.e., *pressure* and *force*, interchangeably throughout the *thesis*.

Chapter 2

Force Input and the Bidirectional Problem

With the introduction of the *Apple*™ *iPhone* in 2007, *multi-touch* has become available to the *general public* and has proven its applicability ever since [Chang et al., 2010]. As a result, *mobile devices*, like *smartphones* have replaced traditional *desktop computers* for many people and are used in a *variety* of different *contexts*. However, operating these *devices* in *encumbered situations* often requires *one-handed use*, facing *reachability issues*, especially when using *larger phones* like the *iPhone 7* or *iPhone 7 Plus*. In addition, due to the *thumb's physical size*, small targets are *occluded*. To tackle both of these issues, *force input* can provide an *additional dimension* and complement *multi-touch interaction* to allow *value manipulations* from a static location.

One-handed use of *mobile devices*, like *smartphones*, faces *reachability issues*.

The following *chapter* presents an *application example* to illustrate how *pressure-based input* can overcome *reachability-*, as well as *occlusion issues* within a *podcast application*. In addition, an *overview* of the *thumb's anatomy*, along with resulting *challenges* for *one-handed smartphone use*, are discussed. *Force input* is presented as possible *solution*, followed by the *three major components* of *pressure-based interaction*, namely *transfer function*, *selection-* and *pressure-control-mechanism*. Finally, the *chapter* concludes with the *bidirectional problem* along with resulting *research questions*, including the *necessary steps* to answer them.

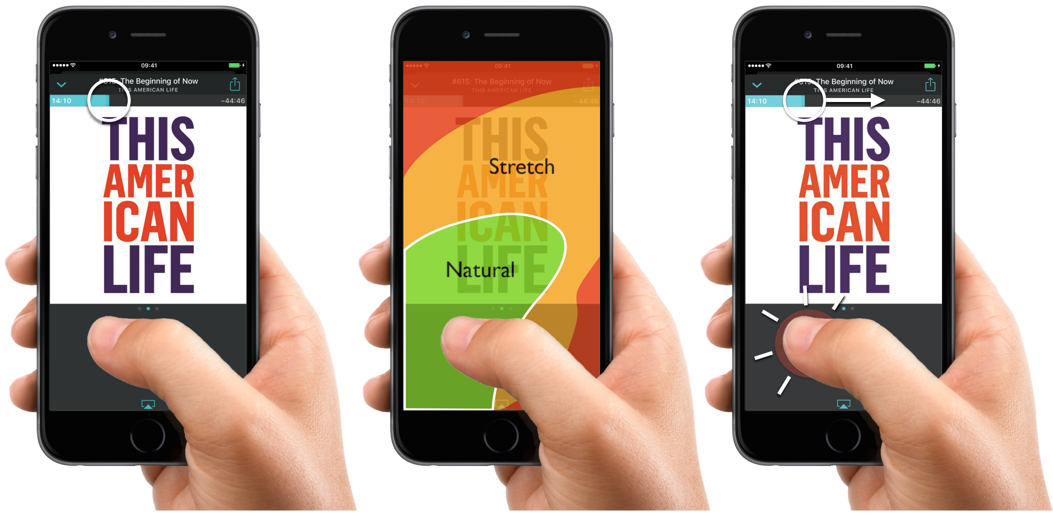


Figure 2.1: Reachability problem within a *podcast application*: Left: *slider* (coupled to current *playback position*) located out of *thumb's reach*, Middle: limited *thumb's interaction range*, indicated by *colored contour lines* (*green*: easy to reach, *red*: difficult to reach), Right: *value manipulation* using *pressure-input* from a static location [Hurff, 2017, Chicago Public Media, 2017, Overcast Radio LLC, 2017].

2.1 Application Scenario

The following *scenario* describes a typical *use case*, how *pressure input* can overcome *reachability* and *occlusion issues* within a *podcast application*: While standing at the *bus stop*, a *user* wants to listen to an *audio podcast*, e.g., a new episode of *This American Life* [Chicago Public Media, 2017]. However, she has already listened to the *current topic* and wants to *skip* a bit *ahead*. Unfortunately, her *left hand* is currently holding a *linen bag*, carrying *books* she wants to return to the local library. Hence, the *slider* that is coupled to the *current playback position* is located out of her *thumb's reach*, since she is operating her smartphone *single-handed* and only using her *thumb* (figure 2.1, *left*).

Some *areas* of the *smartphone*, especially the *top-left* and *bottom-right corner*, are more *difficult* to reach than others.

This *reachability problem* can be visualized by work of Hurff [2017], who used *colored contour lines* (*figure 2.1, middle*) to indicate that some *areas* of the *smartphone* are more difficult to reach than others. In this regards, *green* represents *regions* that are *easily accessible* to the user's *thumb*, while *orange* and *red* refer to *areas* that are *more difficult* to reach [Hurff, 2017].

To overcome these *reachability issues*, *force input* can be used to *apply pressure* to the *play-button* in order to adjust the *current playback position* (*figure 2.1, right*). This way, the *user's thumb* can stay within its *comfortable interaction range* and does not need to operate the *slider* directly. In addition, *occlusion issues* are avoided.

2.2 Challenges of *One-Handed Interaction*

The *accessible interaction range* during *one-handed use* is limited to the *thumb's workspace*.

Having stated an *application example* how *force input* enables *value manipulations* from a *static location*, it is important to note that the restriction of the available *interaction range* to the *thumb's workspace*, causes a *key limitation* of *one-handed smartphone use* [Hirotaka, 2003]. This is because *remaining fingers* are required to *stabilize* the *phone*. As a result, some *areas of the smartphone* are *more difficult* to reach than others. Subsequently, we briefly refer to the *thumb's anatomy* and *resulting limitations* during *one-handed smartphone use*. In this regards, we identify *two major challenges*, namely the *reachability problem* and *visual occlusion* for which *pressure-based input* can provide a *possible solution*.

2.2.1 Anatomy of the *Human Thumb*

Single-handed *smartphone-interaction* heavily relies on the *movement* and *interaction capabilities* of the *human thumb*. Although a deep analysis of its *mechanics* is beyond the scope of this *thesis*, a basic understanding of its *ergonomics* and *characteristics* is crucial, to identify *possible limitations* of *one-handed use*. Thus, the following *section* provides a brief *overview* of the *thumb's anatomy*.

The *human hand* comprises the *carpus* as well as *digits*, including *fingers and thumb*.

According to Schwarz [1955], the *human hand* is built from a *set of bones* that are partitioned into the *carpus*, defining the *human's wrist*, as well as the *digits*, including *fingers* and *thumb*. This way, the *carpus* comprises *eight bones*, namely *greater multangular*, *navicular*, *lunate*, *triquetrum*, *pisiform*, *lesser multangular*, *hamate* and the *capitate* [Schwarz, 1955].

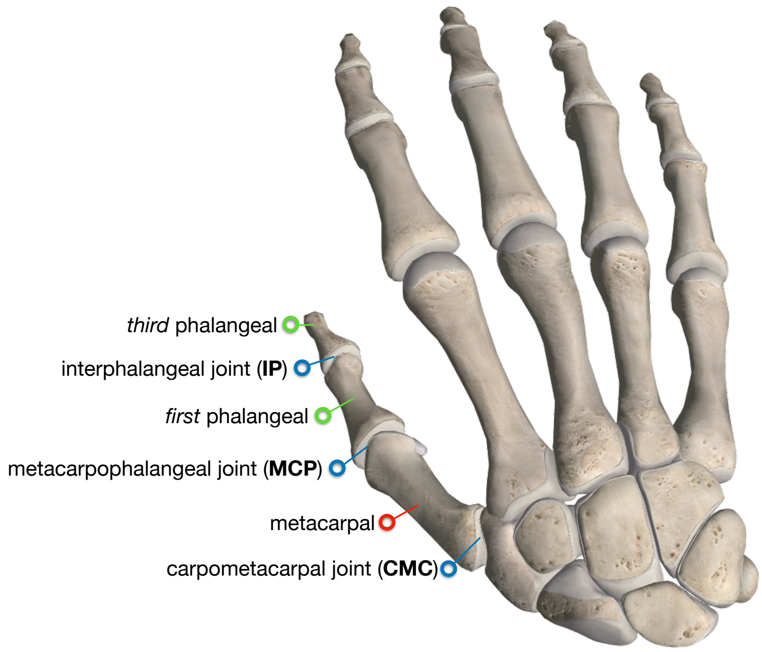


Figure 2.2: Three degrees of freedom of the *human thumb*: *interphalangeal joint* (IP), *metacarpophalangeal joint* (MCP) and *carpometacarpal joint* (CMC) [3D4Medical, 2017].

The *thumb* features three degrees of freedom, offered by the IP-, MCP- and CMC-joint.

In contrast, *digits* consist of *metacarpal*, as well as *phalangeal segments* and maintain a *similar structure*, except for the *thumb* (figure 2.2). This way, the *thumb* does not contain a *second phalangeal*, but offers *greater flexibility* in the *carpometacarpal joint*. Hence, it features *three degrees of freedom*, provided by the *interphalangeal-* (IP), *metacarpophalangeal-* (MCP) and *carpometacarpal-joint* (CMC) [Schwarz, 1955].

As a result, the *thumb's interaction workspace* within the *three dimensional space* is defined by the *thumb's ability* to reposition *phalanges* for a fixed position of the *carpometacarpal joint*. This way, the *thumb* supports *flexion* and *extension*, i.e., *movements* within 45-60° opposing the *palmar plane* [Schwarz, 1955], as well as *abduction* and *adduction*, i.e., *motions* opposite or in direction of the *second metacarpal* [Trudeau et al., 2012b]. Although these *characteristics* provide the *thumb* with *versatile movement capabilities*, it also suffers from *limitations* within *one-handed use*, namely *reachability-* as well as *occlusion issues*.

Figure 2.3: Left: repositioning of the *slider* (trivial solution). Right: *bottom right-hand corner* still *difficult* to reach; *natural area* too small to fit all *user-interface controls* [Hurff, 2017, Chicago Public Media, 2017, Overcast Radio LLC, 2017].

2.2.2 Reachability Problem

Having stated the *thumb's movement capabilities* in the previous section, we subsequently refer to the *first challenge*, i.e., the *reachability problem*, users encounter when operating *smartphones* single-handed and only using their *thumb*. According to Karlson and Bederson [2007], the *thumb's interaction range* is characterized by the *length*, *strength* and *mobility* of the *user's thumb*. As a result, *individual differences* can lead to *reachability issues* during *one-handed use*. Note that the *reachability problem* even becomes more important with the *increasing size* of today's *smartphones* [Karlson and Bederson, 2006a]. In this regards, maintaining the *device* in a *stable position* causes *physical stress* and limits the remaining *interaction capabilities* of the *user's thumb*.

The thumb's *movement capabilities* are characterized by its *length*, *strength* and *mobility*.

Referring to the *application scenario* as stated above, a *naive solution* to the *reachability problem* would reposition the *slider* to be located within the *thumb's reach* (figure 2.3, *left*). However, note that reaching the *bottom right-hand corner* remains *difficult* and requires *awkward grip changes* (figure 2.3, *right*).

Force offers an *additional dimension* to the *ones* offered by *multi-touch*.

In addition, due to the *limited size* of the *thumb's interaction range*, there is not enough *space* to fit all *UI-elements* within reach of the *user's thumb* (figure 2.3, *right*). To tackle these *issues*, *pressure-input* adds a *third dimension* to the ones offered by *multi-touch* and allows to assign *more interactions* to the same *small area*. Consequently, *additional functionality* can be triggered by *exerting* a *sufficient amount* of *pressure*, matching *predefined levels*. Hence, in case of our *application example*, *pressure variations* result in *adjustments* of the current *playback position*.

It is important to note that *pressure-based interaction* does not only address the *reachability problem* in context of *one-handed smartphone-interaction*, but also tackles the issue of *visual occlusion*, as caused by the *thumb's physical size*.

2.2.3 Visual Occlusion

Visual occlusion depicts the *second challenge* of single-handed smartphone use.

In contrast to the *previous section*, *one-handed smartphone-interaction* not only suffers from *reachability issues*, but also from *visual occlusion*, causing *usability issues* within *one-handed use*. This way, the *intended target* is *occluded* from *visual gaze*, as soon as its *diameter* is exceeded by the *thumb's physical size* [Vogel and Baudisch, 2007]. As a result, *users* do not know about their *ongoing interaction* due to the lack of *continuous feedback*. Note that this *issue* is also referred to as *fat-finger problem* and represents the *second challenge* of *one-handed smartphone-interaction* [Vogel and Baudisch, 2007].

To overcome this *limitation*, *pressure-based interaction* allows to decouple the *control* from the *intended target*, and perform *value manipulations* from a static location. Consequently, *visual occlusion* is avoided, since the *user's thumb* can stay within its *comfortable interaction range*, and does not need to interact with the *control* directly.

Having stated both *challenges* of *one-handed smartphone use* and how *pressure-input* can provide a *possible solution*, the following section deals with *pressure characteristics* and its importance for the field of *human computer interaction*.

2.3 Pressure Characteristics

Everyday tasks like *object manipulations* depend on people's ability to *sense* and *exert* various amounts of *force* [Johansson and Flanagan, 2009]. As a result, *pressure-based input* represents a *familiar input modality* that is *suited* to be used within the field of *human computer interaction* (HCI). This way, it is capable of augmenting traditional *smartphone-interaction* to overcome *limitations* of *one-handed use*. The following *section* deals with the *main characteristics* of *pressure-based input* and provides an *overview*, about what has to be considered, when using *force* as *input modality*.

Humans are *well-familiar* with *force-input*.

Pressure Controllability According to Johansson and Flanagan [2009], *human fingers* and *thumb* contain *tactile afferents* that are responsible to provide *detailed information* about the *intensity*, *direction* and *spread* of *contact pressure* during *object manipulations*. These *details* are required by the *human brain* to perform *action-planning* and respond to *unexpected* outcomes. As an example, *grabbing* a cup of tea requires *load-forces* to *lift* the cup from its underlying *surface*, as well as *grip-forces* to *maintain* the cup's *stable position* [Johansson and Flanagan, 2009]. While the *brain* relies on *visual cues* to provide *initial estimates* about the amount of necessary *force*, *values* are *adjusted* as soon as *feedback* from *tactile afferents* is received [Johansson and Flanagan, 2009]. Hence, *people* exhibit profound *pressure-control capabilities*.

Tactile afferents provide *information* about the *intensity*, *direction* and *spread* of *contact pressure*.

Number of Pressure-Levels Moreover, it is important to note that *pressure-based input* can be either used in a *discrete-* or *continuous fashion*. This way, 8-10 *discrete pressure-levels* can be distinguished, given that *visual feedback* about the ongoing *interaction* is provided [Shi et al., 2008, Ramos et al., 2004, Cechanowicz et al., 2007]. In contrast, although *continuous input* supports a possibly *infinite number of levels*, they remain *finite* due to *usability issues*. Note that the *amount* of *distinguishable levels* is crucial, since it specifies the *achievable bandwidth* when using *force* as *interaction modality* [McLachlan et al., 2014].

Humans can control 8-10 *distinct pressure levels*.

Force is not always applied towards the *finger's pointing direction*.

Direction, Variation, Accuracy In addition, *force* is usually applied in the *direction*, in which the finger *initially makes contact* with the *force-sensitive device* [Herot and Weinzapfel, 1978]. In this manner, the *direction* remains *constant*, even if the *user's fingers* are rotated. That's why, *pressure* is not always *applied* towards the *finger's pointing direction* [Herot and Weinzapfel, 1978]. Equally important, *maintaining pressure* is found to be *difficult*, since *force* is sensible to *small deviations* [Herot and Weinzapfel, 1978]. Finally, even though *force* does not convey a *feeling* for a virtual objects's *physical weight* and *size* [Herot and Weinzapfel, 1978], it shows *promising results* in terms of *accuracy*, given that *visual feedback* is provided [Herot and Weinzapfel, 1978].

People's pressure-control capabilities are affected by *environmental conditions*.

Environmental Impact Please be aware that different *environmental conditions* may result in *inadvertent pressure variations* [Stewart et al., 2012]. In this regard, *walking* is found to have a *strong impact* on *user's ability* to *control pressure*, since it causes *more errors*, *longer selection times* and *higher cognitive load* [Wilson et al., 2011]. In addition, *people* tend to *exert more pressure* while *walking* compared to a *sitting condition*. According to Stewart et al. [2012], also *other conditions* like *weather*, *terrain* or the *emotional state* of the *user* may lead to *undesired force fluctuations*. Nevertheless, *multi-touch* is revealed to yield *poor performance* in *physical demanding situations*, like *carrying shopping bags*. In contrast, *force* does not require *significant changes* in *grip* or *hand posture*, since the *user's thumb* can stay within its *comfortable interaction range* [Feng et al., 2015]. In addition, *force input* does not rely on *accurate pointing*, as required by *multi-touch*, and hence allows for *eyes-free-interaction* if combined with a *non-visual feedback modality* [Wilson et al., 2011].

Transience bundles *two main properties:*

- *natural inverse*
- *bounce back*

Transience Finally, *force* is characterized as being *transient*, i.e., any *pressure exertion* is inevitable followed by *pressure release* [McLachlan et al., 2014]. Note that *transience* bundles *two properties*, namely *natural inverse* and *bounce-back* [Ghazali and Dix, 2005]. This way, *natural inverse* specifies *pressure's characteristic* to offer an *action* that allows to *undo* any *outcome*, previously produced. In contrast, *bounce back* denotes the *ability* to return to the *starting condition*, as soon as *pressure* is no longer applied [McLachlan et al., 2014]. In this manner, *pressure* returns to *zero-force* by visiting any *state* in between. Above-stated *characteristics* have shown that people are *well-familiar* with *pressure-based input*. That's why this *interaction modality* is well-suited to be used within the field of *human computer interaction*. The *following section* draws attention to the *three fundamental components* that are required for *pressure-based interaction* before turning to the *bidirectional problem* that depicts *pressure's major limitation*.

2.4 Pressure-based Interaction

Force input requires *three components:*

- *transfer function*
- *pressure-control mechanism*
- *selection mechanism*

Pressure-based interaction relies on *three essential components*, namely *transfer function*, *pressure-control-* and *selection-mechanism*. While the *transfer function* is responsible for providing a *consistent mapping* between *value-* and *force-sensitive range*, a *pressure-control mechanism* determines how *force* translates to *value manipulations*. Finally, the *selection mechanism* is used to complete the *user's choice* and *pick* one of the *values* from the *application domain*. Subsequently, we refer to each of these *components* and clarify their *importance* in context of *pressure-based input*.

2.4.1 Transfer Function

The *transfer function* strongly affects *people's performance* with *force input*.

The *transfer function* depicts the *first component* and *processes* input that is provided by a *force-sensitive resistor* (FSR). In this way, *pressure exertion* lowers the *resistance* of the utilized *sensor*, producing *raw data* about how much *force* is *applied* [Darbar et al., 2016]. To be able to *utilize* this *data*, the *transfer function* is responsible for mapping *measurements* of the FSR, i.e., distinct *pressure-levels*, to *values* of the *application domain* [Stewart et al., 2010]. According to Ramos et al. [2004], choosing an *appropriate transfer function* is crucial, since it has a *strong impact* on *people's performance*. As a result, *recent work* in the *area* has come up with a *variety* of different *transfer functions* where each is dedicated to *specific needs* of the *application domain*.

Transfer functions:

- *linear*
- *low-centered quadratic*
- *fisheye*
- *logarithmic*
- *parabolic-sigmoid*

In addition to a *linear mapping* [Ramos et al., 2004], a *quadratic transfer function*, *centered* around the *lower part* of the *pressure-range* can account for *negative influences* caused by *force variations* [Cechanowicz et al., 2007]. In this regard, the majority of *levels* of the *force-sensitive-range* are assigned to *lower values* of the *application domain*. Moreover, a *fisheye-function* is resistant against *small deviations* and *stabilizes* the *user's selection* by magnifying the *area of interest* [Shi et al., 2008]. Similar to the *quadratic-mapping*, a *logarithmic function* provides *better control* when *little force* is encountered [McCallum et al., 2009]. Finally, a *parabolic-sigmoid function* is *less sensitive* at the *lower-* and *upper-part* of the *force-sensitive range* while following almost a *linear-mapping* in between [Ramos and Balakrishnan, 2005].

The sensor's output should be linearized first, before choosing the *transfer function*.

Although several *transfer functions* exist, it is important to note that the *choice* of the *proper one* heavily relies on the *properties* of the utilized *sensor* [Stewart et al., 2010]. In this regards, many FSRs do not follow *linear resistance*, but rather produce *output* that is more sensitive to *deviations*. Hence, as suggested by Stewart et al. [2010], an *op-amp based current-to-voltage converter circuit* should be used to *linearize* the *sensor's output* first, before choosing one of the *transfer functions*. In addition, Stewart et al. [2010] found a *linear-mapping* to *perform best*, when the *sensor's output* is *linearized*.

2.4.2 Pressure-Control Mechanism

There are *two categories* of *pressure-control mechanisms: positional-* and *rate-based control*.

Apart from the *transfer function*, as mentioned in the *previous section*, a *pressure-control mechanism* depicts the *second component* of *pressure-based interaction*. This way, it is responsible for *appropriate value adjustments* in response to the amount of *force* that is *currently applied*. According to Ng and Brewster [2016], *pressure-control mechanisms* can be partitioned into *two main categories*, namely *positional-* as well as *rate-based control*.

Positional control assigns *force* to an *absolute position* within the *application's value range* [Wilson et al., 2011]. In this manner, *values* at the *top*, require *significantly more force* than *values* at the *bottom*. In addition, to *keep* the *current selected*

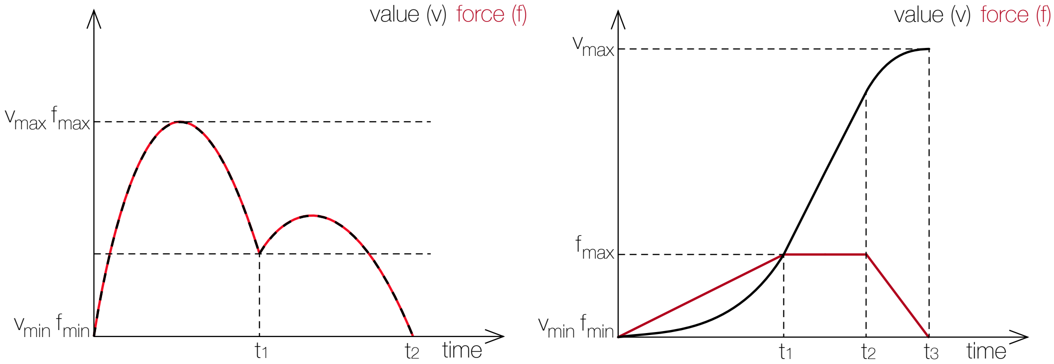


Figure 2.4: *Pressure-Control Mechanisms*: Left: *Positional Control* - value and force correspond to each other. Right: *Rate-based Control* - force specifies, how fast the value changes.

value, people have to *maintain pressure* at the *corresponding level*. However, one of the *main advantages* of *positional control* is that it offers *immediate access* to *overshoot corrections* [Wilson et al., 2011]. This way, *pressure* can be released at any time, causing the *cursor* to return to its *original location*.

Positional control features simple *overshoot corrections*.

By contrast, *rate-based control* couples *force* to the *speed* with which *values* are changing [Wilson et al., 2011]. As a result, the *cursor* moves with the *speed* that is defined by the amount of *force* that is *currently applied* until *pressure* is fully released again [Wilson et al., 2011]. Please be aware that *rate-based control* does not feature *overshoot corrections*, as they are possible for *positional control*. This is because the *cursor's movement speed* is only specified towards a *single direction* [Wilson et al., 2011].

Rate-based control does allow to *correct overshoots* out of the box.

2.4.3 Selection Mechanism

Finally, a *selection mechanism* depicts the *last component* of *pressure-based interaction*. Even though *pressure's continuous nature* does not suggest an *obvious mechanism* [McLachlan et al., 2014], researchers have come up with *several solutions* that provide *workarounds* to *trigger events* or *pick values* out of the *application domain*. *Figure 2.5* provides an *overview* of *selection mechanisms* by referring to *time-pressure diagrams* along with the *point* in *time* the *selection* is made.

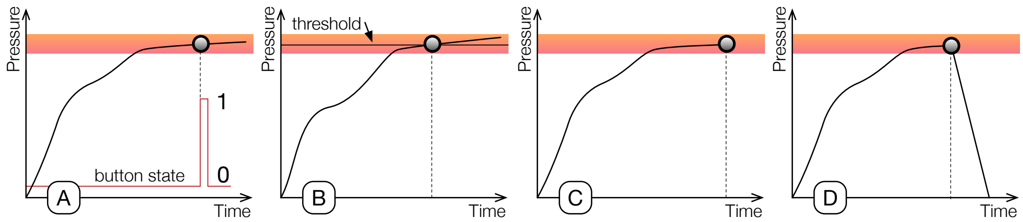


Figure 2.5: *Pressure Selection Mechanisms*: A: *Click* - pressing a button, B: *Threshold* - crossing predefined thresholds, C: *Dwell time* - maintaining pressure for a predefined time interval, D: *Quick release* - quickly releasing pressure [Ramos et al., 2004]

Click requires a button to finalize the *user's selection*.

Threshold triggers events if *force-levels* are exceeded.

As illustrated in *figure 2.5* (A), *Click* provides a *trivial solution* by using an *additional button* to complete the *user's selection* [McLachlan et al., 2014]. However, note that operating a *button* while *maintaining pressure* remains difficult. In contrast, as illustrated in *figure 2.5* (B), *Threshold* triggers *discrete events* as soon as *predefined thresholds* are exceeded [McLachlan et al., 2014]. For instance, one could think of a *drawing application* in which *line-thickness* is adjusted according to how much *force* is applied. Consequently, *thicker lines* are accomplished by *crossing associated levels*.

Dwell time suffers from *artificial delays*, and requires to *keep moving* if *selections* are not yet desired.

On the contrary, as shown in *figure 2.5* (C), completing one's choice with *Dwell-time* requires to *maintain pressure* for a *predefined duration*, e.g., *one second*. Even though *dwell-time* is found to achieve *high accuracy*, *two drawbacks* are identified [McLachlan et al., 2014]. *First*, *interactions* suffer from *artificial delays*. *Second*, users have to keep *moving* if *selections* are not yet desired [McLachlan et al., 2014].

Predicting the *value*, users intended after *force* was quickly released, is a *challenging task*.

Finally, *Quick-release* (*figure 2.5, D*) offers *fluent transitions* without causing *major delays* [McLachlan et al., 2014]. Nevertheless, please be aware that selections are *error-prone*, since predicting the *intended value* depicts a *difficult task*, after *pressure* was *quickly released* [McLachlan et al., 2014].

Having stated the *three major components* that are necessary when using *force* as *input modality*, the following *section* deals with the *bidirectional problem* that must be considered, since it *limits* the *applicability* of *force input* in the area of *human computer interaction*.

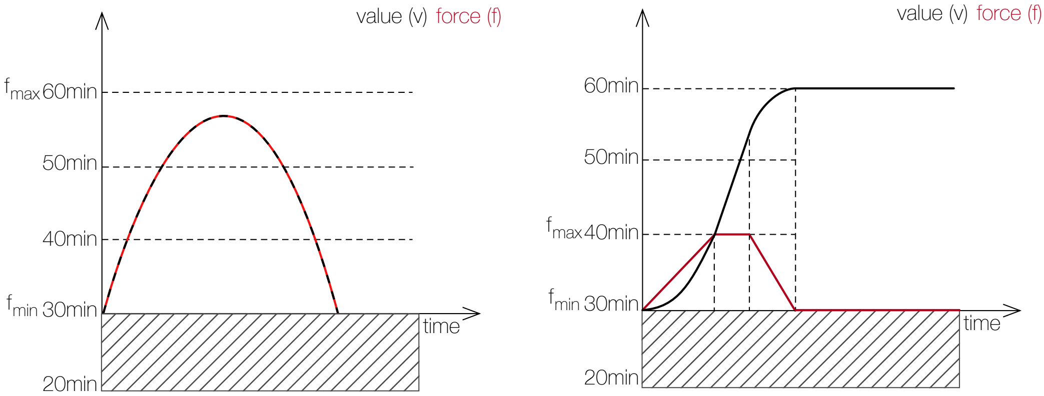


Figure 2.6: Bidirectional Problem: Values below the *current value* ($30min$) cannot be reached (*crossed area*) using either *positional-* (left) or *rate-based control* (right).

2.5 The Bidirectional Problem

Even though *pressure* has been used for several tasks like *line-thickness control*, *mode-switching* or *discrete menu selection*, it is a *one-way continuous parameter* [Rekimoto and Schwesig, 2006, Mandalapu and Subramanian, 2011]. As a result, *pressure-based input* does not support *bidirectional value manipulations* that are required for many common tasks, like *zooming*, *scrolling* or *brightness control* [Rekimoto and Schwesig, 2006, Mandalapu and Subramanian, 2011]. To demonstrate this *bidirectional problem*, and to motivate why there is a *need* for *bidirectional force input*, an illustrative example is given as follows:

Force input does not support *bidirectional value manipulations.*

Revisiting the *application example* from *section* 2.1, a *user* wants to adjust the *current playback position* within a *podcast application*, using *pressure-based input*. This way, *force* can be *applied* to manipulate the *value's absolute position* using *positional control* (figure 2.6, left), or the *value changing speed* in case of *rate-based control* (figure 2.6, right). However, even though *positional control* features simple *overshoot corrections* that allow to return to the *original location* (figure 2.6, left, $57min$ to $30min$), *values* below $30min$ cannot be reached (figure 2.6, *crossed area*). This is because *values* cannot be further *decreased* as soon as *pressure* is *fully released*. This *limitation* is *crucial*, since it prevents *users* from skipping to *playback positions*, they have already listened to.

Because of the *bidirectional problem*, the *playback position* within a *podcast application* can not be set to *values* below $30min$.

Similarly, even in case of *rate-based control*, *values* below the *original location*, i.e., $30min$, cannot be accomplished (figure 2.6 *crossed area*). Hence, *value manipulations* are only possible within a *single direction*. In this regards, *users* can *slowly apply pressure* to navigate with *increasing speed* (figure 2.6 *right*, $30min$ to $40min$), or keep the *pressure level* unchanged to move with *constant acceleration* (figure 2.6 *right*, $40min$ to $52.5min$). Nevertheless, even though *pressure release* results in *slower manipulations* (figure 2.6, right, $52.5min$ to $60min$) where *zero-force* corresponds to *zero-speed*, the *value's movement direction* remains unchanged.

Values below the *original location* can not be accomplished.

As a result, independent of the utilized *control mechanism*, *pressure-based interaction* is limited in the way that *values* below the *original location* cannot be accomplished (figure 2.6 *crossed area*). This behavior is problematic, especially in context of GUI controls that require *value manipulations* in *both directions*, e.g., when *adjusting* the *current playback position* within a *podcast application* [Mandalapu and Subramanian, 2011]. That's why providing an *appropriate solution* to the *bidirectional problem* is crucial to make *force input* applicable to more *application domains* [Spelmezan et al., 2013a], and hence depicts the *primary objective* of this *thesis*.

Primary objective: overcome the *bidirectional problem*.

Having referred to the *bidirectional problem* as *major limitation*, the following *section* deals with resulting *research questions* as well as the *necessary steps* to answer them.

2.6 Research Questions

In this chapter, *force input* has been identified as promising *input modality* to complement *multitouch* within *one-handed smartphone use*, because of the following reasons: First, *pressure input* can address *reachability*- as well as *occlusion issues* that are caused by the restricted *movement capabilities* of the *human thumb*. Second, *pressure* offers an *additional dimension* such that the *user's thumb* can stay within its *comfortable interaction range* and perform *value manipulations* from a *static location* [McLachlan et al., 2014]. Finally, *more interactions* can be assigned to the *same small area*, without affecting the device's *form factor* [McLachlan et al., 2014].

However, even though people have profound *pressure control capabilities* [Johansson and Flanagan, 2009], *pressure* is limited in the way that it is *unidirectional* [Spelmezan et al., 2013a]. Hence *pressure-based interaction* suffers from constraints posed by the *bidirectional problem* that needs to be *alleviated* to make *pressure-based input* applicable to more *application domains*.

According to Rekimoto and Schwesig [2006], tackling the *bidirectional problem* is a *non-trivial task*. Hence this *thesis* follows a *systematic procedure* to find an *appropriate solution* according to the following *research questions*:

Finding an *appropriate solution* to the *bidirectional problem* is a *non-trivial task*.

R1 What are potential *interaction designs* to enable *bidirectional force input* from a *static location*?

R2 Which design *performs best* and is *most preferred*?

In this regards, *Chapter 4* deals with the *first research question* and takes a *glimpse* at the *design space* of *bidirectional force input*, considering the *human thumb's movement capabilities*. This way, several *components* are identified, namely *pressure-control mechanism*, *pressure mapping* as well as *direction mechanism*, from which *eighteen bidirectional interaction designs* are derived.

To answer the *second research question* and evaluate the *design's appropriateness* in terms of *user preference* and *performance*, *Chapter 5* contains the *first study* and adapts a *target acquisition* and *selection* task as used by Heo and Lee [2012], Ramos et al. [2004] and Shi et al. [2008]. This way, *participants* are asked to select *predefined targets* as *quickly* and *accurately* as possible, using several *bidirectional designs*. Moreover, the *second study* (*Chapter 6*) concentrates on *remaining designs* of the *previous study*, by comparing their *performance* against a *baseline condition*.

A *target-acquisition* and *selection-task* is used to answer the *second research question*.

Having described the *main objective* of this *thesis* along with resulting *research questions*, the following *chapter* provides the *necessary background knowledge* by referring to *related work* in the *area* of *thumb ergonomics* as well as *pressure-based interaction modalities*.

Chapter 3

Related Work

Operating *smartphones* in *encumbered situations*, like *carrying books* within a *linen bag*, limits *users' interaction possibilities* to *single-handed device operation*. In this regards, *usability* suffers from *reachability issues* that are especially problematic in context of *larger phones*, since the available *interaction range* is limited to the *thumb's workspace* [Hirotaka, 2003]. However, even though *placing UI-controls* within *reach* of the *user's thumb* would provide a *trivial solution*, there might not be *enough space* to fit all *controls* due to the area's limited size. Moreover, interacting with *controls* directly also faces *occlusion issues*, caused by the *physical dimensions* of the *user's thumb* [Vogel and Baudisch, 2007]. To counteract these limitations within *one-handed use*, *force input* adds a *third dimension* that allows *value manipulations* while resting the *thumb* at a *static location*. Still, *force* is limited in the way that it is *unidirectional* [Spelmezan et al., 2013a].

Finding a *solution* to the *bidirectional problem* requires knowledge about:

- *thumb ergonomics*
- *pressure-based interaction modalities*

Nevertheless, *solving* the *bidirectional problem* does not only require an understanding of *pressure-based interaction*, but also *knowledge* about *thumb ergonomics*, since only the *user's thumb* is available within *one-handed use* [Hirotaka, 2003]. That's why *bidirectional interaction designs* have to account for the thumb's *limited movement capabilities* to ensure *comfort* within *one-handed use*. Hence, the *following section* draws attention to *recent work* in the area of *thumb ergonomics* and refers to several *pressure-based interaction techniques* categorized by their *application domain*.

3.1 Thumb Ergonomics of Single-Handed Smartphone Interaction

Karlson and Bederson [2006a] revealed that *one-handed use* is preferred if the *interface* allows for *single-handed smartphone operation*.

Karlson and Bederson [2006a] explored *user-preference* regarding *number of hands*, and examined how *one-handed interaction* affects the thumb's *movement performance* in an *empirical evaluation*. Results indicate that participants prefer *one-handed use*, and only rely on *multiple hands* if the *interface* does not allow for *single-handed interaction* [Karlson and Bederson, 2006a]. Referring to *thumb ergonomics*, the *device's form factor* is identified to have a *strong impact* on *movement performance* [Karlson and Bederson, 2006a]. This way, even though *ergonomics* are found to be *best* within the *center* of the device, *long-distance targets* along with *very close located ones* are difficult to reach, since they require *awkward hand postures* that are *difficult* to maintain [Karlson and Bederson, 2006a]. These *findings* are interesting, since they suggest that *one-handed use* is preferred if the *interaction designs* would better account for *single-handed device operation*.

Trudeau et al. [2016] identified that *people* perform *faster* when *both hands* are used.

Similarly to work presented by Karlson and Bederson, Trudeau et al. [2016] investigated the human thumb's *movement capabilities* within *one-* and *two-handed* smartphone use. This way, *participants* performed a *repetitive tapping task* by *acquiring* several *targets* in alternation [Trudeau et al., 2016]. Note that *data* about *hand-* as well as *thumb-kinematics* is gathered using a *motion tracking system*, to quantify the thumb's *movement performance* according to *Fitts' Law* [MacKenzie, 1995]. However, in contrast to *user-preference results* by Karlson and Bederson, *findings* reveal *superior performance* of *two-handed-use*. This way, utilizing *both hands* led to 9% greater *performance* in terms of *Fitts' law*, 7% faster *movement times*, as well as 4% higher *precision* [Trudeau et al., 2016]. In addition, *bimanual interaction* resulted in *less physical stress*, since *one* hand is dedicated for *holding the device*, while the *other* is responsible for *interacting* at the *front* [Trudeau et al., 2016]. Unfortunately, the *authors* only examined *tapping* and did not consider other *movements* like *panning* or *sliding*. Still, *results* indicate that the thumb's *movement capabilities* are more *limited* within *one-handed use*. This motivates us to consider *thumb ergonomics*, when creating *bidirectional pressure-based interaction designs*.

Figure 3.1: Movement capabilities of the Human Thumb: *abduction/adduction*: movements *opposite* or *in direction* of the *second metacarpal* [Trudeau et al., 2012b], *extension/flexion*: movements within $45\text{-}60°$ opposing the *palmar plane* [Schwarz, 1955] [California State University, 2017].

Note that these findings are consistent with previous results by Trudeau et al. [2012b] where they identified the *bottom right-* as well as the *top right corner* of *mobile devices* to be *most difficult* to access by the *human thumb*. As a result, *interface designers* should avoid placing *UI-controls* within areas that require the *thumb* to operate at its *limits* in *flexion* and *extension*, i.e., at the *extrema* within the *plane* parallel to the *palm* [Trudeau et al., 2012b]. This way, *controls* should rather be *positioned* at the phone's *middle-left* or *top-right* to be *easily accessible* without *loosing performance* [Trudeau et al., 2012b].

The *top-right* and *bottom-right corner* is *difficult* to reach by the *thumb* [Trudeau et al., 2012b].

In a *follow up work*, Trudeau et al. [2012a] also analyzed the *thumb's movement performance* with respect to *movement direction*, *orientation* and *device size* within *one-handed use*. In this manner, several *orientations* based on the *cardinal directions*, i.e., *north* (N), *east* (E), *south* (S) and *west* (W), along with several *device sizes*, i.e., *small*, *flip*, *large* and *pda*, are evaluated within a *repetitive tapping task* [Trudeau et al., 2012a]. Note that *thumb operations* within NE $\leftrightarrow$ SW and N $\leftrightarrow$ S depend on the *carpometacarpal joint's abduction* and *adduction* (*section* 2.2.1, *figure* 3.1), whereas *movements* within NW $\leftrightarrow$ SE and E $\leftrightarrow$ W require *flexion* and *extension* (*figure* 3.1) [Trudeau et al., 2012a]. *Results* suggest that the *combination* of *inner movements* with an *increased device size* poses *physical constraints* on the *CMC joint's movement*

Movements within NE $\leftrightarrow$ SW and N $\leftrightarrow$ S require *abduction* and *adduction* while movements within NW $\leftrightarrow$ SE and E $\leftrightarrow$ W rely on *flexion* and *extension*.

Movements within NE ↔ SW require *less degrees* of *freedom*.

capabilities, and hence limits the *thumb's interaction range* [Trudeau et al., 2012a]. Interestingly, in comparison to all other *directions*, *movements* within NE ↔ SW *performed best*. Trudeau et al. explained this result by the *observation* that movements within NE ↔ SW rather depend on *abduction* and *adduction* than *flexion* and *extension* and hence require less *degrees of freedom*, compared to movements within NW ↔ ES orientation [Trudeau et al., 2012a]. Findings motivate us to consider a *rolling gesture* when creating *bidirectional interaction designs* to respect the *thumb's movement performance* along its *natural axis*, i.e., NE ↔ SW. In this regard, *abduction* and *adduction* led to *better performance* than *flexion* and *extension* [Trudeau et al., 2012a].

Flexion and *extension* result in *slower movements* compared to *abduction* and *adduction* [Xiong and Muraki, 2014].

Please be informed that *findings* by Trudeau et al. are obtained for *mobile devices* using *physical keys*. Thus, it remains *uncertain* whether *results* do also apply to *touchscreen-enabled devices*, like *smartphones*. To mitigate this problem, Xiong and Muraki [2014] investigated *thumb ergonomics* on *smartphones* using *tapping-*, *moving-* as well as *circling tasks*, where participants had to *tap buttons* of various *size*, *move* within *multiple locations* using *abduction/adduction* or *flexion/extension*, or perform *clockwise-* as well as *anti-clockwise* motions respectively [Xiong and Muraki, 2014]. It is worth mentioning that the *authors* also applied *electromyography* (EMG) to analyze the *thumb's muscle activity* for respective *tasks* [Xiong and Muraki, 2014]. Findings confirmed *previous results* of Trudeau et al. that *flexion* and *extension* lead to *slower movements* than *abduction* and *adduction* [Xiong and Muraki, 2014]. Equally important, *smaller targets* are identified to result in *less performance* and significantly higher *muscle activity* than *larger ones* [Xiong and Muraki, 2014]. However, *circling motions* did not have an impact on *movement performance* [Xiong and Muraki, 2014]. Consequently, *previous results* do also apply to *touchscreen-enabled devices*.

Moreover, research by Campos et al. [2014] also focussed on *touchscreen-enabled devices* within *one-handed use* and provides a *heat-map* that categorizes the *phone's interaction area* according to *thumb ergonomics*. This way, a *discomfort index* is calculated with respect to a *comfort position*, considering the *absolute difference* within *euler angles* of the *thumb's joints* [Campos et al., 2014]. Note that the *reference position* represents *maximum comfort*, and hence features a *zero-discomfort index*. Consequently, the *index increases*, as soon as *slight deviations* from the *comfort position* are recognized [Campos et al., 2014]. Note that the resulting *heat-map* provides *guidance* to *application developers* to consider *thumb ergonomics* when deciding about the proper *placement* of *user interface controls*, and hence mitigate *reachability issues* [Campos et al., 2014]. Unfortunately, the *authors* did not calculate *discomfort indices* for *multiple gestures*. Findings would have allowed us to *gain insights* about which movements are *best* to *specify directions* within *bidirectional interaction designs* in terms of *thumb ergonomics*.

Roudaut et al. [2009] created a *gesture-set*, containing *swiping-*, *dragging-*, *rubbing-*, and *rolling-gestures* that is designed to consider the *thumb's limited movement capabilities*.

Finally, Roudaut et al. [2009] followed a different approach to examine *thumb ergonomics* within *one-handed use*. This way, the authors developed a *gesture set*, containing *swiping-*, *dragging-*, *rubbing-*, as well as *rolling-gestures* that are especially designed to consider the *limited movement capabilities* of the *human thumb* [Roudaut et al., 2009]. *Gestures* are beneficial, since they allow to extend the *thumb's interaction possibilities* without having to show *toolbars* or *context menus* that are difficult to manage within *one-handed use* [Roudaut et al., 2009]. To evaluate the *gesture's performance*, *participants* were asked to perform *each gesture* within a *predefined area*. Findings are promising, since *recognition rates* achieved an *overall accuracy* of 95.3% [Roudaut et al., 2009]. Interestingly, *rolling gestures* are found to be *faster* in *cardinal* ($230ms$) than *circular directions* ($339ms$), performed *quicker* than *rubbing* ($938ms$) and *dragging* ($458ms$) and also are *most preferred* by participants [Roudaut et al., 2009]. These *findings* can be explained by the fact that *rolling* offers *immediate access* to *different commands*, without including *artificial delays* [Roudaut et al., 2009]. Note that the authors demonstrated the *gesture's applicability* by mapping *rolling directions* to well-known commands, like *cut*, *copy* and *paste* and achieved *higher efficiency* than *toolbars* and *context menus* [Roudaut et al., 2009]. Results encourage us to consider *rolling* in context of *force input* to offer *immediate access* to *both directions* within *bidirectional interaction designs*.

Roudaut et al. [2009] identified that *rolling-gestures* are *well-suited* for the *thumb* within *one-handed use*.

Previous results have emphasized the importance of *thumb ergonomics* within *one-handed use*. However, please be aware that *bidirectional interaction designs* also require knowledge about how *force* can serve as *interaction modality*. Consequently, the *following section* deals with *recent work* in the area of *pressure-based interaction techniques* categorized by their *application domain*.

3.2 Pressure-based Interaction Modalities

Although *force* is limited in the way that it is *unidirectional*, and hence suffers from *constraints* posed by the *bidirectional problem*, *recent work* has come up with several *pressure-based interaction techniques* that have proven the *appropriateness* of *force input* in multiple *domains*. Thus, the following *sections* provide an *overview* about *pressure-based interaction modalities* within *four* different *domains*, namely *multi-touch/tablet-*, *mouse/keyboard-*, *pen-based-* as well as *mobile device-interaction*.

3.2.1 Multi-touch/Tablet Interaction

Note that the *first domain* focusses on *properties* of *touch-sensitive tablets*, and investigates how *pressure-based interaction* can overcome *limitations* of *multi-touch input*.

Force input allows to distinguish multiple states *using* distinct pressure intensities.

In this manner, Buxton et al. [1985] identified the *lack* of *multi-touch* to be missing the ability to *trigger events* while *fingers* are moving. Hence, *interactions* are *less expressive* than conventional *mouse interaction*, where *pointing* and *selections* happen simultaneously [Buxton et al., 1985]. Even though using additional *function keys* would provide a *possible solution*, it would requires *both hands* and consequently does not allow for *single-handed interaction*. To tackle this issue, Buxton et al. [1985] proposed to exploit *force input* to trigger *multiple states* using *predefined pressure intensities*. This way, *soft pressure* can be used for *target acquisition*, i.e., *tracking*, while *stronger pressure* confirms the *user's selection* [Buxton et al., 1985]. Note that the authors demonstrated their proposed *pressure-based interaction technique* using *two variants* of a *drawing application*.

In the *first variant*, *light pressure* provides feedback about the user's *drawing location*, while *stronger pressure* causes *ink* to become visible on-screen [Buxton et al., 1985]. In contrast, the *second variant* uses *continuous pressure* to allow *immediate access* to *line-thickness adjustments* [Buxton et al., 1985]. However, please be aware that *pressure exertion* over *long distances* is found to be *exhausting* due to *friction* causing *usability issues* [Buxton et al., 1985].

Pressure application over *long distances* is exhausting [Buxton et al., 1985].

In contrast to work by Buxton et al., Forlines et al. [2005] stated that *input devices* using *direct manipulation*, like *multi-touch tablets*, do not require an additional *tracking state*, since *target acquisition* is already done, as soon as the *user* starts *interacting* with the device [Forlines et al., 2005]. As a result, the *authors* assigned *pressure-input* to *additional functionality* and proposed *Glimpse*, a *pressure-based interaction modality* that facilitates exploration of *modification possibilities* with comfortable *undo* [Forlines et al., 2005]. Note that an *undo-mechanism* is essential to support *people* to try out *modifications* without having to worry about that *changes* cannot be *undone* [Forlines et al., 2005]. However, *undo* is often hidden within *application menus* and takes *time* and *effort*, since it is usually not part of the *interaction cycle*. That's why any *improvements* regarding *undo* are desired [Forlines et al., 2005]. To cope with this issue, *Glimpse* introduces an *additional state*, where *light pressure* allows *previewing changes*, while *stronger pressure* commits any *adjustments* that are currently made [Forlines et al., 2005]. Note that *uncommitted changes* are undone at any time as soon as *pressure* is *fully released*. The presented *interaction modality* represents a *promising use case* of *pressure-based input*, since it has already found its way into *today's smartphone interaction*. This way, *Apple*™ adopted this *technique* with their *recent introduction* of *peek and pop* [Apple©, 2017c].

Glimpse exploits *force input* to allow *previewing* of *changes* with *comfortable undo* [Forlines et al., 2005].

ForceDrag depicts another *pressure-based interaction technique* within *multi-touch/tablet interaction* and provides a *solution* for missing *modifier-keys* that are *heavily used* within *desktop environments* [Heo and Lee, 2012]. Note that even though *modifier-keys* can be *virtually simulated*, they result in *less screen space* for *content presentation* [Heo and Lee, 2012]. To overcome this *flaw*, Heo and Lee introduced *ForceDrag* that exploits *force input* to specify *dragging modes* using *predefined*

Heo and Lee [2012] proposed a *force-lock mechanism* to deal with *friction* for *long-distance targets*.

pressure intensities [Heo and Lee, 2012]. According to Heo and Lee, *ForceDrag* is especially beneficial in context of *3D-applications* where it provides *users* with the *ability* to *move*, *rotate* or *scale* arbitrary objects [Heo and Lee, 2012]. In addition, the *authors* developed a *force-lock mechanism* that tackles the issue of *friction*, as noted by Buxton et al.. This way, *users* can stay within the *same mode* by specifying *desired pressure-levels* beforehand, and hence do not need to *maintain pressure* over *long distances* [Heo and Lee, 2012]. Unfortunately, *specifying modes* using *pressure-based input* requires *selection mechanisms*, like *dwell time* (*section* 2.4.3), causing *artificial delays* [Heo and Lee, 2012]. In addition, please be aware that the *force-lock mechanism* does not support *mode changes* while *moving* [Heo and Lee, 2012]. Still, the *author's findings* are promising, since they provide a *possible solution* to mitigate *issues* caused by *friction* along *far away targets*.

Rendl et al. [2014] proposed a *refined version* of the *force-lock mechanism* proposed by Heo and Lee [2012].

Finally, *Presstures* depicts *force-augmented multi-touch gestures* that are designed to obtain *less cluttered* UIs [Rendl et al., 2014]. In this manner, *Presstures* do not require *visual feedback*, since they only rely on *user's pressure perception* [Rendl et al., 2014]. Note that Rendl et al. adapted the *force-lock mechanism*, as presented by Heo and Lee [2012], but refined it to work with *multi-touch gestures*. This way, *mode-selection* is only allowed within an *area* of $1.5cm$ around the *initial contact position* to provide a *seamless transition* to the *remaining part* of the *gesture* [Rendl et al., 2014]. Equally important, *force variations* are measured until the *predefined area* around the *initial contact position* is *exceeded*, yielding a *target pressure* that matches the *maximum* among all *measured levels* [Rendl et al., 2014]. Interestingly, *findings* revealed that *Presstures* could only be *efficiently controlled* when used with *two pressure levels* [Rendl et al., 2014]. Note that Rendl et al. explained these *results* by identifying *thresholds* as not being appropriate for *mode selection*, since *participant* seem to have individual *pressure perceptions* [Rendl et al., 2014]. Hence, we decided to equip *bidirectional interaction designs* with *continuous feedback* about the amount of *exerted pressure* to alleviate *usability issues* due to *individual differences*.

Having stated *pressure-based interaction techniques* within *multi-touch/tablet* interaction, the following *section* focusses on *corresponding modalities* in the *area* of *keyboard* and *mouse*.

3.2.2 Mouse and Keyboard Interaction

Several attempts have been made to utilize *force input* in a *wide range* of *application domains* due to its *versatile characteristics*. While the *previous section* has focussed on *pressure-based interaction modalities* in the area of *multi-touch/tablet* interaction, this *section* deals with *recent attempts* to utilize *force* to enhance *mouse* and *keyboard interaction*.

PressureFish follows a rather *simple approach* and attaches a single *force sensitive resistor* to *traditional mice* [Shi et al., 2008]. In this regards, the *authors* utilized *positional control* (*section* 2.4.2) for *discrete menu selections* [Shi et al., 2008]. Unfortunately, the *authors* did not look into *bidirectional value manipulations*, since *values* could not be *further decreased*, as soon as they have returned to their *original location*. However, *PressureFish* represents an *interesting approach*, since it is found to achieve *great accuracy* if combined with a *fisheye-discretization function* [Shi et al., 2008].

Shi et al. [2008] proposed *PressureFish*, a *pressure-augmented mouse* that uses a *fish-eye transfer function*.

In addition, Cechanowicz et al. [2007] presented *guidelines* to equip *traditional mice* with *pressure-sensing capabilities* and explored *several techniques* that exploit *force* for *value selection* [Cechanowicz et al., 2007]. In this regard, *pressure-sensors* should be placed in range of the *user's fingertips*, but should not interfere with the *interaction range* of the *index finger*, since it is reserved for *traditional mouse interaction* [Cechanowicz et al., 2007]. As a result, the *mouse's top* is identified to be *well-suited* to control *lower force*, while the *mouse's left* is found to be more *appropriate* for *higher pressure* [Cechanowicz et al., 2007]. In addition, Cechanowicz et al. proposed two *pressure-based interaction techniques*, namely *switch-to-refine* and *tap-and-refine* that are especially designed for *dual-pressure equipped mice* [Cechanowicz et al., 2007]. Note that *switch-to-refine* utilizes the *primary sensor* to allow *coarse-level adjustments*, while the *secondary sensor* allows more *fine-level control* [Cechanowicz et al., 2007]. Finally, *selections* are made using *click-to-select* (*section* 2.4.3). In contrast, *tap-and-refine* offers different *granularities* using the same *pressure-sensor*. This way, *tapping* is used to iterate through *coarse-level values*, while *regular force* results in *fine-level adjustments* [Cechanowicz et al., 2007].

Cechanowicz et al. [2007] proposed *tap-and-refine* and *switch-to-refine*, i.e., *two pressure-based interaction modalities* to allow *coarse-* as well as *fine-level adjustments*.

Even though, *tap-and-refine* offers multiple *levels of precision* through a *single sensor*, it is meant to be used with *dual-pressure equipped mice*, to allow *value manipulations* in *both directions* [Cechanowicz et al., 2007]. Consequently, the *presented approach* is intriguing, since it tackles the *bidirectional problem*, using *multiple sensors*.

Strips exploit *force input* and can be configured as *slider*, *buttons* or *discrete-* as well as *continuous-spinning wheel* [Blaskó and Feiner, 2004].

Turning to **keyboard interaction**, *modalities* resemble each other, since they all exploit *pressure-based input* to assign *more functionality* to a *limited space*. In this manner, Blaskó and Feiner [2004] presented *Strips*, i.e., *four finger-sized regions* that are located on a *pressure-sensitive pad* [Blaskó and Feiner, 2004]. Note that these *areas* do not require *homing* such that *fingers* stay rested at the *input device* [Blaskó and Feiner, 2004]. In addition, each *strip* can be *divided* into *multiple subregions* and can be *configured* to allow *various interactions*. This way, *Strips* can be used as *linear slider*, *dynamically resizing buttons* or *discrete/continuous spinning wheel* [Blaskó and Feiner, 2004]. To be able to assign multiple *designs* to the *same strip*, the *authors* utilized *force input*, and developed a *technique* called *pop-through* that allows to *double* the *number of strips* and *step* through *associated interactions* [Blaskó and Feiner, 2004]. Moreover, a *dual-finger mechanism* adds *three virtual strips* between each of the *four physical ones*. As a result, the *number of available strips* can be *virtually increased* from 4 to a total of $2 \times (4+3) = 14$ [Blaskó and Feiner, 2004]. Unfortunately, it remains uncertain whether *users* can control *Strips* with *reasonable speed* and *precision*, since the *presented technique* has not been *tested* in an *empirical evaluation*. Still, *Strips* are inspiring, since they eliminate the need for *on-screen widgets* and hence reduce *visual clutter* when *screen space* is limited.

Pressure-text utilizes *force input* to eliminate the need for *repetitive key-presses*, as required by *multi-tap typing* [McCallum et al., 2009].

Similar to the *previous approach*, *PressureText* also exploits *pressure's ability* to assign *additional functionality* to *restricted areas*, but includes *force-sensors* into each *individual key* of a *mobile phone* to reduce *repetitive key-presses* as caused by *multi-tap typing* [McCallum et al., 2009]. In this regard, up to *four characters* can be *distinguished* using predefined *pressure intensities*. Note that the *presented modality* achieved *similar performance* compared to *multi-tap* [McCallum et al., 2009]. This way, *PressureText* (9.1wpm) performed *faster* than *multi-tap*(8.64wpm) [McCallum et al., 2009].

Finally, *One-press control* depicts the *last modality* within this *section* and integrates *force input* into the *control cycle* of *desktop-class keyboard interaction* [de Jong et al., 2010]. Besides *regular-key events* that maintain the *keyboard's basic functionality*, *additional events*, namely *medium-* and *hard-repeat*, are attached to notify *software applications* about *pressure intensities* [de Jong et al., 2010].

One-press control exploits *force input* to replace *complex key-combinations* with *single-key events* [de Jong et al., 2010].

According to de Jong et al., *possible use cases* include the *replacement* of *complex key-combinations* by *single-key events*, e.g., $[alt]+F4 \rightarrow [hardRepeat]+F4$, as well as interactive *preview/exploration*-capabilities [de Jong et al., 2010]. Results suggest that *One-press control* led to 7.4 out of *ten successful trials*, and is learnable within approx. $15min$ [de Jong et al., 2010]. Unfortunately, *the authors* did not look into *bidirectional force input* and only utilized *pressure* to distinguish *multiple states*.

From *above-stated examples* we conclude that *force input* is *heavily researched* in context of *multi-touch/tablet* as well as *keyboard/mouse interaction*. Nevertheless, note that *force input* is also adapted by other *domains*, like *pen-based-* or *mobile-device interaction*. Hence, we provide the *reader* with an *overview* of the *wide range* of *application domains* by *first* referring to *pressure-based interaction modalities* in context of *pen-based interaction*.

3.2.3 Pen-based Interaction

Please be informed that *research* regarding *force input* in context of *pen-based interaction*, is dominated by *work* conducted by Ramos et al.. In this manner, the *authors* proposed *several pressure-based interaction techniques* to operate *multi-state widgets*, i.e., *PressureWidgets*, with the aid of *visual feedback* [Ramos et al., 2004]. Note that *PressureWidgets* are characterized by the property (e.g., *position*, *angle* or *scale*), *pressure* is mapped to, as well as the *widget's visual elements* in form of *cursor* and *targets* [Ramos et al., 2004]. As a result, the *authors* proposed *four* different *widgets*, as illustrated in *figure* 3.2, that utilize *force* as *input modality* in context of *pen-based interaction*:

Ramos et al. [2004] proposed *PressureWidgets*, i.e., *force-enabled widgets* for *pen-based interaction*.

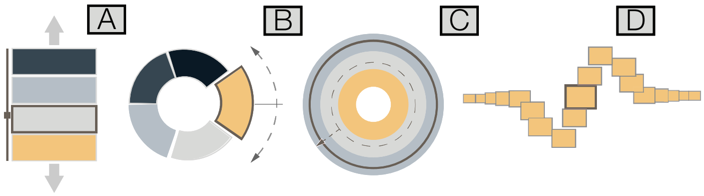


Figure 3.2: Pressure Widgets: A: *flag-widget*, B: *rotating-expanding-pie-widget*, C: *bullseye-widget*, D: *twist-lens-slider-widget* (*figure* adapted according to [Ramos et al., 2004]).

First, the ***flag-widget*** (*figure* 3.2, A) utilizes *force* to *move* a list of *items* under a *cursor* [Ramos et al., 2004]. Interestingly, the *authors* decided to choose a *static-* rather than a *dynamic-cursor*, since they came to the conclusion that keeping the *cursor* static would afford *pressure application* and deter *users* from *moving* the *stylus* [Ramos et al., 2004]. Unfortunately, Ramos et al. did not further evaluated this *hypothesis*.

In contrast, the ***rotating-expanding-pie*** (*figure* 3.2, B) represents the *second widget* and arranges *targets* in a *circular shape*, rather than a *sequential list* [Ramos et al., 2004]. This way, *pressure* is coupled to the widget's *rotating angle*, such that the *widget* rotates as soon as *force* is applied. Note that *continuous feedback* about the *user's current selection* is provided by increasing the *selected target's scale* [Ramos et al., 2004]

Third, the ***bullseye-widget*** (*figure* 3.2, C) utilizes a *ring cursor* and adjusts its *scale* according to different *pressure intensities*. This way, the *cursor* expands, as soon as *more force* is applied, while *pressure release* results in a *reduction* of the *rings diameter* [Ramos et al., 2004].

Finally, the ***twist-lens-slider widget*** (*figure* 3.2, D) also controls a *sequential list* of *items* via *force input*, similar to the *flag-widget*, but uses a *fisheye-visualization* where *pressure* is coupled to the *cursor's scale* [Ramos et al., 2004]. Consequently, the *list expand* as soon as *force* is exerted [Ramos et al., 2004].

Unfortunately, *PressureWidgets* have not been assessed using an *empirical evaluation*. Still, they draw attention to the *importance* of *continuous feedback* in context of *pressure-based input*. Hence, we are encouraged to include *continuous visual feedback* for *bidirectional interaction designs*.

Zliding exploits *force input* to enable *high-precision parameter manipulations* within *pen-based interaction* [Ramos and Balakrishnan, 2005].

In a *follow-up work*, Ramos and Balakrishnan [2005] introduced *Zliding*, a *pressure-based interaction technique* that seamlessly integrates *zooming* and *scaling* to facilitate *high precision parameter manipulations* in context of *pen-based interaction* [Ramos and Balakrishnan, 2005]. Note that the concept of *Zliding* is instantiated by the *Zlider-Widget* that features *adjustable granularity* using *force input* [Ramos and Balakrishnan, 2005]. This way, *users* can choose *coarse granularity* for *initial value manipulations* and switch to *fine-level adjustments* when *precision* is required. In addition, *scroll zones* are included at the *slider's extreme points* to allow *continuous scrolling* within the *value range* [Ramos and Balakrishnan, 2005].

PressureWidgets and *Zliding* highlight the *importance* of *visual feedback* in context of *force input* [Ramos and Balakrishnan, 2005, Ramos et al., 2004].

Similar to *PressureWidgets* as stated above, *Zliding* includes *visual feedback* about the *current selected value* using a *red line* as well as a *pressure-cursor* that indicates *pressure intensities* through *color variations* [Ramos and Balakrishnan, 2005]. Interestingly, the *authors* included a *clutching-mechanism* that behaves similar to *force-lock*, as presented by Heo and Lee [2012]. This way, *pressure release* after leaving the *slider's area* locks the *current granularity* that *further increases* as soon as *force* is *reapplied* [Ramos and Balakrishnan, 2005]. *Findings* revealed that *Zliding* allows *high precision parameter manipulations*, but suffers from *unintended zoom operations* during *dragging operations* [Ramos and Balakrishnan, 2005]. That's why, Ramos and Balakrishnan proposed to *temporarily* disable *scale adjustments* while *dragging*.

Finally, *PressureMarks* deals with the *issue* of *selection-action tasks*, to get *sequential structures*, even though the *original task* is meant to be *performed* in parallel [Ramos and Balakrishnan, 2007]. Note that these *tasks* are commonly used within *pen-based interaction* and suffer from *artificial delays* caused by *consecutive executions* [Ramos and Balakrishnan, 2007]. To mitigate this *issue*, Ramos and Balakrishnan proposed to assign *unique signatures* to *pressure intensities* to

PressureMarks suggest to assign *unique signatures* to *force variations* [Ramos and Balakrishnan, 2007].

trigger *actions* and *selections* in parallel [Ramos and Balakrishnan, 2007]. In this regard, a *parsing algorithm* analyses the *movement* of the *pen* in addition to *force variations* to recognize up to *four different signatures* [Ramos and Balakrishnan, 2007]. Results indicate that *PressureMarks* are *easily learnable* and perform 27% *faster* compared to recent *serial selection-action methods* like *lassoing and pigtail* [Ramos and Balakrishnan, 2007]. *PressureMarks* are exciting, since they suggest to use *unique pressure patterns* to specify *directions*.

Finally, we draw the *reader's* attention to *pressure-based interaction modalities* in context of *mobile device interaction* and discuss several *solution candidates* to the *bidirectional problem* that have been proposed in this *area*.

3.2.4 Mobile Device Interaction

Mobile device interaction differs from all *previous domains*, since *mobility* strongly affects *usability* and poses additional *challenges* within *one-handed use* [Boring et al., 2012]. Hence, *research* in this *area* took advantage of *force input* to enhance *frequent tasks* like *zooming*, *panning*, or *scrolling*.

PreSenseII utilizes *differences* in *finger-contact size* to enable *bidirectional value manipulations* [Rekimoto and Schwesig, 2006].

Rekimoto and Schwesig [2006] presented *PreSenseII*, a *novel input device* that features both *touch-* as well as *force-sensing capabilities* to enable *bidirectional value manipulations* [Rekimoto and Schwesig, 2006]. This way, the *authors* utilized *variations* in *finger contact-size* as *mode-indicator* to indicate whether the *current selected value* should be *in-* or *decreased* if *force* is applied [Rekimoto and Schwesig, 2006]. As a result, the *presented modality* allows *bidirectional value manipulations*, like *zooming-in* or *out* in a *map application*, or *scrolling* through a *list of items* with *adjustable speed* [Rekimoto and Schwesig, 2006]. However, please be aware that the *effectiveness* of *force input* strongly depends on whether *feedback* about the *ongoing interaction* is provided [Rekimoto and Schwesig, 2006]. In this regard, the *authors* decided to provide *tactile feedback* about the *current scroll speed* in form of *unique intervals* in which *tactile marks* are induced [Rekimoto and Schwesig, 2006].

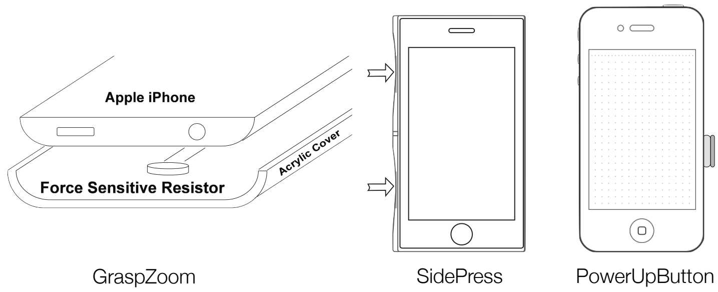


Figure 3.3: *Pressure-based interaction modalities* within *mobile device interaction* [Miyaki and Rekimoto, 2009, Spelmezan et al., 2013a,b]

PreSenseII suggests to include *tactile-feedback* for *bidirectional interaction designs*.

Unfortunately, *PreSenseII* has not been *examined* in a *detailed evaluation*. Nevertheless, this *pressure-based interaction technique* is auspicious, since it provides a *possible solution* to the *bidirectional problem* without having to rely on *multiple sensors*. Moreover, we are reinforced in our decision to include *tactile feedback* for *bidirectional interaction designs*.

GraspZoom utilizes a *front-sliding gesture* to enable *bidirectional force input* [Miyaki and Rekimoto, 2009].

GraspZoom follows a *similar approach* and exploits *force input* to improve *zooming* and *scrolling* within *one-handed use* [Miyaki and Rekimoto, 2009]. However, instead of using a *pressure-sensitive surface* at the front, the *authors* attached a *force-sensitive resistor* underneath an *acrylic cover* to the *back* of the *mobile device* (*figure* 3.3). Interestingly, *GraspZoom* also supports *bidirectional value manipulations* using a *front-sliding gesture* to *specify directions*. This way, *zoom-in operations* only require *force application* at the *back*, while *zoom-out operations* are preceded by the *front-sliding gesture* [Miyaki and Rekimoto, 2009]. Even though *GraspZoom* has not been evaluated in an *empirical evaluation*, it is a *promising modality*, since it allows *continuous scrolling* from a *static location* [Miyaki and Rekimoto, 2009].

Moreover, *SidePress*, as illustrated in *figure 3.3*, follows the *same objective* as *GraspZoom*, but makes use of *two pressure-sensors* (*figure* 3.3), attached to the *side* of a *mobile device* [Spelmezan et al., 2013a]. In this manner, the *authors* try to mitigate *occlusion issues* of *finger-scrolling* within *document navigation* [Spelmezan et al., 2013a].

GraspZoom enables *bidirectional force input* through *multiple sensors*, attached to the *side* of *mobile devices* [Spelmezan et al., 2013a].

Note that *force variations* correspond to *unique interaction events* that can be mapped to *multiple actions*, depending on the *application domain* [Spelmezan et al., 2013a]. As an example, *light-*, *strong-* as well as *max-click events* are available that allow to *move* to the *next/previous line*, *next/previous page* or to the *last/first page* respectively [Spelmezan et al., 2013a]. Despite *promising results* that identified *SidePress* to be *more efficient* than *touch* for *long-distance scrolling*, it still seems to be awkward to apply *pressure* at the *side* of *mobile devices*. Hence we are encouraged to look into *alternative approaches* to enable *bidirectional force input* from a *static location*.

The *Power-Up button* combines *force-* with *proximity sensing* to overcome *both parameter's limitation* of being *unidirectional* [Stewart et al., 2012].

In a *follow-up work*, Spelmezan et al. provided an *alternative solution* to the *bidirectional problem* in form of the *Power-Up Button* (*figure* 3.3) that combines *force-* with *proximity-sensing* [Spelmezan et al., 2013b]. Indeed, both *input modalities* are limited in the way that they are *unidirectional* [Spelmezan et al., 2013a]. Nevertheless, considering *both* in *combination* not only allows to trigger *discrete-up* and *down-events*, but also *continuous input* by *approaching* or *leaving* the *button's area* [Spelmezan et al., 2013b]. This way, *users* can provide *input* using *six* distinct events, namely *click*, *quick-release*, *discrete-up/down* and *continuous-up/down* [Spelmezan et al., 2013b]. Note that the *authors* demonstrated the *applicability* and *potential* of their *approach* by *controlling* any kind of *widget*, using only the *Power-Up button* [Spelmezan et al., 2013b]. Although the *presented modality* is *encouraging* in the sense that it combines *force-* with *proximity sensing*, it lacks *helpful guidance* to assist *novices* to learn the *set* of *gestures*.

ForceEdge exploits *force input* to determine *scroll-speed adjustments* [Antoine et al., 2017].

Finally, *ForceEdge* exploits *force input* to *facilitate autoscrolling* when *screen space* is limited [Antoine et al., 2017]. Unlike *standard techniques*, *ForceEdge* does not specify *scroll-speeds* according to the *distance* from the *device's edges*, but rather analyzes *force variations* to determine *scroll-speed adjustments* [Antoine et al., 2017]. In this regard, *ForceEdge* involves *three steps*: *First*, the *interaction* is initiated by *grabbing* an *object* that should be *moved* to a *distant location*, *positioned* outside of the *view's boundaries* [Antoine et al., 2017]. *Second*, the *object* is moved towards the *view's bottom edge* into a *predefined area*. *Finally*, the *object's new location* is specified through *force input* that controls the *speed* with which the *underlying content* is moved [Antoine et al., 2017].

As a result, the *control area* can be significantly *smaller* compared to *conventional approaches* and requires *less movement*, since *force* is applied from a *static location* [Antoine et al., 2017]. Interestingly, the *authors* evaluated the *presented modality* using a *scrolling task* where *objects* had to be *moved* as *quickly* and *accurately* as possible [Antoine et al., 2017]. Although *ForceEdge* was found to be 58% *faster* and 16% *more accurate* than *standard techniques*, the *authors* only studied *top-to-bottom scrolling* and did not examine *other directions* [Antoine et al., 2017]. Still, the *concept* seems to be applicable for *bidirectional scrolling*, since *movements* in the *opposite direction* would only require *control areas* at the *remaining edges*. However, please be aware that the *thumb's movement performance* might *differ* when moving in the *opposite direction* [Antoine et al., 2017].

ForceEdge revealed *promising results* in terms of *speed* and *accuracy* [Antoine et al., 2017].

The *previous sections* should have raised the *reader's awareness* for the *many attempts* that have been made to utilize *force input* in *various domains*. However, capturing them all is beyond the *scope* of this thesis. Still, the *following sections* provides *pointers* into *additional domains* that exploit *force input* as *interaction modality*.

3.2.5 Further Directions

Recent work also examined *force input* to overcome *occlusion issues* within *smartwatch interaction* (*section* 2.2.3). This way, *BandSense* attached *pressure-sensors* to the *lower-* and *upper-part* of the *watchband* to minimize the *need* for *multi-touch input* [Ahn et al., 2015]. Consequently, *users* can perform *tapping* as well as *flicking gestures* on their *wristband* in either *horizontal-* or *vertical-direction*. In addition, *continuous input* is provided using *force variations* [Ahn et al., 2015].

BandSense adds *force-sensitive resistors* to the *wristband* [Ahn et al., 2015].

By contrast, *PressTact* uses *four pressure-sensors* at the *sides* of a *smartwatch* to facilitate *occlusion-free interaction* [Darbar et al., 2016]. Interestingly, *sensors* are operated *individually* or any *two* in *conjunction* with *three pressure-intensities*, i.e, *low*, *mid*, *high*, yielding *thirty unique events* that can be *mapped* to a *variety of applications*, like *zoom-in/zoom-out operations*, *image rotation* or *list-selection* [Darbar et al., 2016].

PressTact attaches *pressure-sensors* at the *side* of a *smartwatch* [Darbar et al., 2016].

TactfulCalling exploits *force input* to specify the *level of importance* before placing a *phone call* [Hemmert et al., 2009],

Moreover, *further directions* also include *TactfulCalling* that allows to judge *phone calls* according to their *level of importance* [Hemmert et al., 2009]. In this regard, *Tactful Calling* equips the *caller's phone* with a *force-sensitive dial key* that allows to specify the *call's precedence* before *placing* the *call*. The *technique* is beneficial, since *callers* usually are not aware whether the *callee* is currently engaged [Hemmert et al., 2009]. Consequently, the *callee* can set a *threshold*, up to which *incoming phone calls* are rejected. As a result, *force input* allows to reduce the *amount* of *undesired calls* in *inappropriate situations* [Hemmert et al., 2009].

Finally, *force input* is also used in the *automotive domain*, as demonstrated by *research* conducted by Huber et al. [2016], who obtained a *force interaction language* to trigger *in-car commands*. Interestingly, the *authors* obtained their results from a *controlled experiment* where *participants* were asked to *think aloud* about how they would utilize *force input* to handle *typical in-car operations*, like *air-conditioning/volume control* or *map navigation* [Huber et al., 2016].

Lessons Learned

Current solutions to the *bidirectional problem* combine multiple *input modalities* or *sensors* → need for *bidirectional force input* using a single *force-sensitive resistor*.

This *chapter* has referred to *related work* in the area of *thumb ergonomics* within *one-handed use* and has provided a detailed *overview* about *pressure-based interaction modalities* categorized by their *application domain*. Even though *some attempts* have been made to *provide solutions* to the *bidirectional problem*, the *key issue* of much of this *literature* is that these *approaches* mostly rely on *multiple sensors*, or combine two *unidirectional input channels*, to allow *bidirectional value manipulations*. Consequently, there is a *need* for appropriate *interaction designs* that consider *thumb ergonomics* and enable *bidirectional force input* using a *single force sensitive resistor*.

Please be aware that *findings* in this *chapter* should provide us with the *necessary background knowledge* to *answer* our *research questions* and find an *appropriate solution* to the *bidirectional problem*. Hence, we can now turn to the *ingredients* that are required for *bidirectional interaction designs*, along with the resulting *design space* of *bidirectional force input*.

Chapter 4

Bidirectional Designs

Humans feature *profound pressure control capabilities* to manage *everyday tasks*, like *holding*, *pushing* or *squeezing* an *object* [Stewart et al., 2010]. Indeed, *pressure-sensing* is required to judge an *object's weight* or to determine the *strength* that is necessary to *keep objects* in a *static position* [Stewart et al., 2010]. As a result, *force input* represents an *interaction modality*, with which people are *well-familiar*. Moreover, *force* can augment conventional *multi-touch interaction* with an *additional dimension* that does not require *significant changes* in *hand posture* and allows *continuous input* from a *static location* [Stewart et al., 2010, McLachlan et al., 2014]. Consequently, *force input* is *well-suited* to be used within *encumbered situations* that usually require *one-handed use*.

Recent work in *pressure-based interaction* has failed to come up with an *appropriate solution* to the *bidirectional problem* that uses a single *force-sensitive resistor*.

However, even though these *characteristics* are well understood by *recent work* in the area (*Chapter 3*), and demonstrate the *great potential* of *force input* to mitigate *reachability-* as well as *occlusion-issues* within *one-handed use*, *restrictions* caused by the *bidirectional problem* can not be neglected (*section 2.5*). Nevertheless, to the best of our *knowledge*, *previous work* in this *area* has failed to come up with *dedicated solutions*, since they rather make use of *multiple sensors* or combine of *pressure-* with *proximity-sensing* [Rekimoto and Schwesig, 2006, Spelmezan et al., 2013b]. Hence, there is a need for *dedicated interaction designs* that enable *bidirectional force input* using a single *force-sensitive resistor*.

Aim of this chapter: provide an *answer* to the *first research question*.

At this point, we briefly want to remind the *reader* that the *main objective* of this *thesis* is to find a *solution* to the *bidirectional problem* and make *force input* applicable to *more application domains*. Hence, we decided to build our *research* on a *systematic procedure* that *first* looks at the *components* that are needed to come up with *bidirectional interaction designs*, and *second* conducts a *detailed evaluation* to identify the *technique* that is *most preferred* and *performs best*. Subsequently, we focus on our first *research question*, and propose several *bidirectional designs* that are built from *three essential components*, based on what we have learned from *recent work* regarding *thumb ergonomics* and *pressure-based interaction* (*Chapter 3*).

4.1 Three Essential Components

To promote the *reader's understanding* about how *bidirectional designs* are obtained, this *section* identifies *three essential components* for each of our *designs*, namely *pressure-control mechanism*, *pressure mapping* and *direction mechanism*.

Our *bidirectional designs* can easily be combined with *established selection mechanisms* and *transfer functions* from *literature*.

Please be aware that *bidirectional designs* in fact contain *two additional components*, namely *transfer function* and *selection mechanism*. Nevertheless, considering each of these *components* is beyond the *scope* of this *thesis*. Hence, we point the *reader* to an *increasing number* of *studies* that have already looked at both of these *components* ([Ramos et al., 2004, McLachlan et al., 2014, Cechanowicz et al., 2007, Shi et al., 2008, McCallum et al., 2009, Ramos and Balakrishnan, 2005]), and rather aim for *convenient way* to *specify directions*.

However, to be able to investigate the *suitability* of our *presented designs*, we decided to choose *dwell-time* as *selection mechanism* (*section 2.4.3*), since it is found to achieve *high accuracy* despite causing *artificial delays* [McLachlan et al., 2014]. Likewise, we adapt our *transfer function* to match the *ones* that are *well-established* in *literature*. As a result, we can omit *potential biases* caused by the choice of *transfer function* or *selection mechanism* and focus on *multiple attempts* to *specify directions*. Subsequently, each of the *three essential components* is discussed.

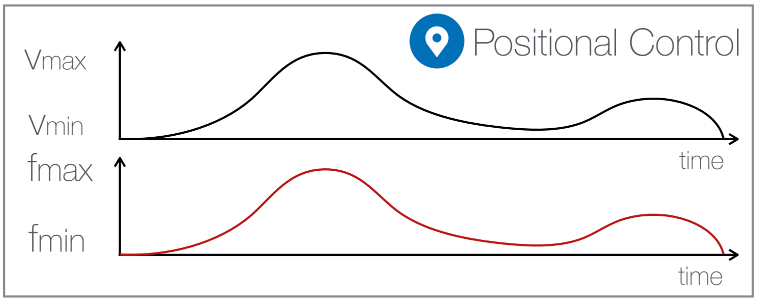


Figure 4.1: Positional Control: *value* and *force* correspond to each other [IOSTE, 2016].

4.1.1 Pressure-Control Mechanism

The *pressure-control mechanism*, as *introduced* in *section 2.4.2*, depicts the *first component* of *proposed bidirectional designs*. Please be reminded that *literature* differentiates between *positional-* as well as *rate-based control* [Wilson et al., 2010].

In *positional-control*, *pressure intensities* are assigned to *absolute positions* within the *value range* [Wilson et al., 2010]. Consequently, as illustrated in *figure 4.1*, *value* and *force* are coupled together. In this regard, as soon as *force* is applied (*red bottom line*), the *value* increases until the *global maximum* is reached (*black top line*). Similarly, the *value* decreases when *force* is *slowly released*, until it reaches a *local minimum* where *pressure* is maintained. Finally, the *value* returns to its *original location* after visiting a *local maximum* in between.

In *positional control* value and *force* are coupled together.

Indeed, *positional control* features simple *over-shoot corrections*, since *users* only have to *release force* to visit *previous locations*. In addition, the *ability* to *decrease values* when *force* is *released*, corresponds to an *intuitive mapping* with which people are already familiar. However, *difficulties* arise if the *value range* contains too many *entries*, since *positional control* is identified to be *less accurate* when exceeding $8-10$ *levels* [Pelurson and Nigay, 2016]. Similarly, *performance* is found to decline if *positional control* is used within *mobile scenarios* [Wilson et al., 2011]. Finally, preserving the *current selection* raises *usability issues*, since *maintaining force* is found to be *difficult* [Ramos et al., 2004].

Positional Control:

+ *overshoot-corrections*
- *less accurate* for >10 levels
- *maintaining force* suffers from *strong deviations*

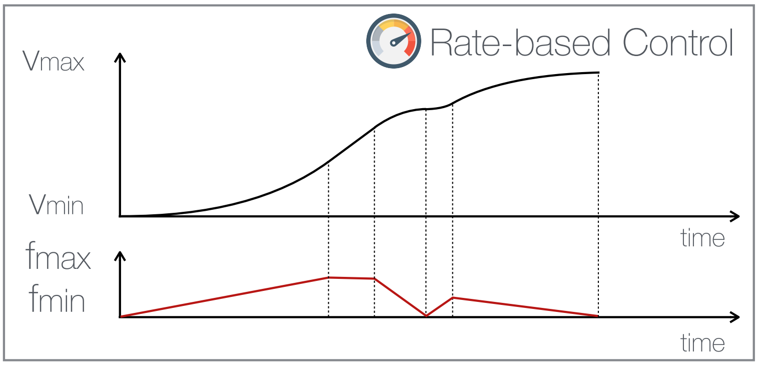


Figure 4.2: Rate-Based Control: *force*$\mapsto$*value-changing speed* (*zero-force* $\hat{=}$ *zero-speed*) [FlatIcon, 2017].

Rate-based control maps *force* to the *speed* with which *values* are changing.

In contrast, *rate-based control* maps *force variations* to the *speed* with which *values* are *changing* [Wilson et al., 2010]. Consequently, as demonstrated by *figure 4.2*, the *value* increases with rising speed, followed by *constant acceleration*, until *force* is *completely released*. As a result, *rate-based control* is not limited to $8-10$ *levels*, but rather offers *control* over a possibly *infinite set* of *distinguishable values* [Pelurson and Nigay, 2016]. In addition, there is no *need* to *maintain force* at *predefined levels*, since the *current value* immediately *stops moving*, as soon as *force* is *completely released*. As a result, controlling the *value's changing speed* rather than the *absolute position*, alleviates *performance issues* of *positional control*, and is identified to be *less mentally demanding* within *mobile scenarios* [Wilson et al., 2011]. Nevertheless, as stated in *section 2.4.2*, *rate-based control* does not support *overshoot corrections*, since *value manipulations* are only possible in a *single direction*.

Rate-based Control:

+ not limited to $8-10$ *levels*
+ no *need* to *maintain force* at *predefined levels*
- *overshoot-corrections*

Indeed, choosing one *mechanism* over the *other* has a *strong impact* on *bidirectional designs' characteristics*, since it determines the *options* that are *available* to *indicate direction*. As an example, using *positional control* suggests that *directions* have to be specified at the *lower part* of the *value range*, since it seems to be rather *difficult*, as soon as *force* is applied. By contrast, *direction changes* in context of *rate-based control* seem to be possible at *any time*, since *users* can linger at an *intermediate location* of the *value range*, without having to return to the *initial location*.

Direction changes within *rate-based control* are possible at any time.

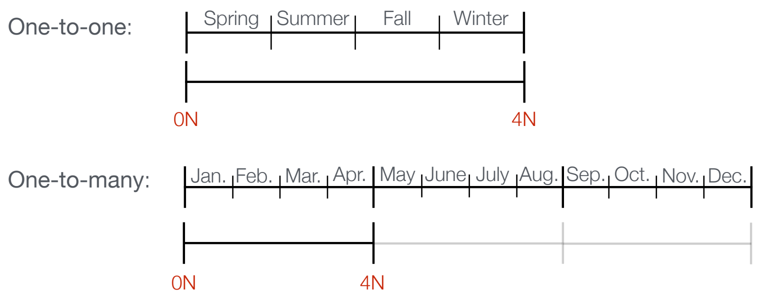


Figure 4.3: Pressure Mappings: *one-to-one: value-* and *force-sensitive range* correspond to each other, *one-to-many: values* are split into *multiple regions*, each assigned to the same *force-sensitive range*.

Consequently, even though *both mechanisms* feature *clear advantages* over each other, there is no *definite choice*. Hence, we decided to explore the *appropriateness* of both *mechanisms* in context of *bidirectional interaction designs*.

There is no definite choice among *positional-* and *rate-based control*.

4.1.2 Pressure Mapping

Moving on to the *second component*, the *pressure mapping* determines how *value-* and *force-sensitive range* are mapped to each other. Note that we distinguish *two mapping-types*, namely *one-to-one* [1:1] as well as *one-to-many* [1:N], whose *differences* are clearly defined by the *example*, as shown in *figure 4.3*. Indeed, as the *name* already suggests, *one-to-one* corresponds to the *mapping* where *value-* and *force-sensitive range* completely coincide. This way, one could think of a *finite set* of *values*, like the *four seasons' names*, that partition the *value range* into *four* different *categories*, each corresponding to a *predefined area* of the *force-sensitive range*.

One-to-one ≙ *value-* and *force-sensitive range* completely correspond to each other.

Conversely, *one-to-many* splits the *value-range* into *multiple regions*, and assigns the same *force-sensitive range* to each of the *segments*. For instance, when using *force input* to select *months* out of the *set* of *twelve possible values*, applying a *one-to-one* mapping would result in *usability issues*, since according to *literature*, as stated in *Chapter 3*, *controlling* more than $8-10$ *levels* results in *reduced performance*.

One-to-many ≙ the *value-range* is split into *multiple regions*, each assigned to the same *force-sensitive range*.

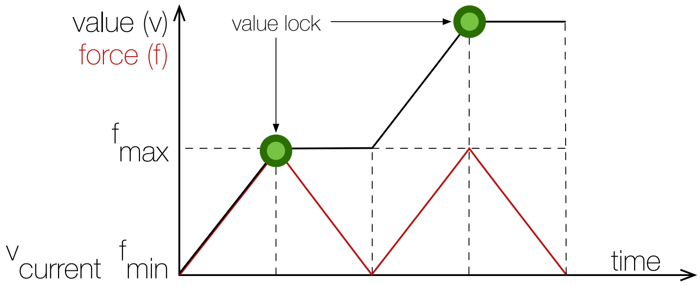


Figure 4.4: Concept of *Positional Pumping:* If *maximum force* is applied, *values* get *locked*, such that *force* can be *fully released* without affecting the *current selected value*.

The *attentive reader* will have already *become aware* of an *important detail* that has to be considered for *positional-control* within *multiple regions*. This way, it is yet uncertain how *months* are selected within *[1:N]-mappings* that are located beyond the *range* of the *sensor*. To resolve this *issue*, the *following section* introduces the *concept* of *positional pumping*.

Positional Pumping

Positional pumping overcomes the *finiteness* of the *force-sensitive resistor*.

Using *positional control* within *multiple regions*, faces the *issue* that *pressure intensities* beyond the *maximum* of a *force-sensitive resistor* cannot be detected. Hence, motivated by the *clutching mechanism* as presented by Ramos and Balakrishnan [2005], we developed the *concept* of *positional pumping* in which *force* can be *completely released* without affecting the *user's current selection*. Indeed, the *concept* is designed to be used within *[1:N] mappings* that split the entire *value-range* into *multiple segments*. In this regard, *borders* between *adjacent regions* act as *jump-over points* where *values* are *locked* until *force* is no longer applied. As a result, *positional pumping* allows *value navigation* among *multiple regions* using a *single force-sensitive resistor*.

From the *graph*, as shown in *figure 4.4*, it is apparent how *positional pumping* is applied to reach *values* that are located outside the *current region*. Starting from the *current value* ($V_{current}$), *users* can *exert force* until a *border* is reached.

Whenever this is the case, the *value* gets locked, as indicated by the *green dot* in *figure 4.4*, such that *force* does not need to be *further applied*. As a consequence, *users* can *reapply force* to *push forward* into the *following segment*. Please be aware that the *concept* of *positional pumping* is heavily used within this *chapter*, since it is required for *bidirectional interaction designs* that utilize *positional control* as *pressure-control mechanism*. Subsequently, we refer to *direction mechanisms*, as the *last* of the *three essential components* presented in this *chapter*.

Using *positional pumping* users can reapply *force* to *push* in to the *following region*.

4.1.3 Direction Mechanism

Having referred to *pressure-control mechanisms* and *pressure mappings*, this *section* draws attention to the *third* and most important *component* to enable *bidirectional force input* from a *static location*. Clearly, only taking advantage of the *previous components* does not allow to *specify directions*, and hence would limit our *designs* to a *single direction*. To alleviate this *issue*, this *section* presents several *toggle*- and *switch-mechanisms* that consider *thumb ergonomics* as well as findings regarding *force interaction modalities* (*Chapter 3*).

It is important to realize that the *distinction* between *switches* and *toggles* is crucial, since choosing *one* over the *other strongly affects* how *bidirectional designs* are perceived by the *user*. Consequently, while *switches* offer *immediate access* to *both directions*, *toggles* only allow to *alternate* between them. Subsequently, *direction mechanisms*, i.e., *switches* and *toggles*, are discussed that are *especially designed* for *one-handed use*.

Direction mechanisms are partitioned into *switches* and *toggles*.

Switches

Thumb Roll represents the first *switch-mechanism* and offers *immediate access* to *both directions*. As illustrated in *figure 4.5*, *users* can initially rest their *thumb* on-screen and *roll* either *left* or *right* to specify *directions*. Moreover, *force* is *released* while *crossing* the *center* to match the *natural rolling-behavior*. We decided to *choose* this *gesture*, since it was identified to be one of the *most ergonomic ways* to extend the *thumb's input expressiveness* within *one-handed use* [Roudaut et al., 2009].

Thumb Roll offers immediate access to both *directions* by rolling *left* or *right*.

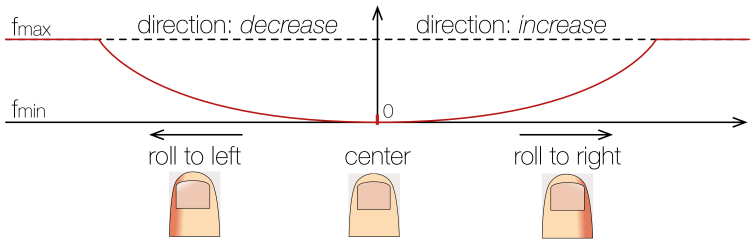


Figure 4.5: Thumb Roll: [rolling left] *decrease*, [center] *none*, [rolling right] *increase*.

Thumb Roll exploits *movements* along the *thumb's natural axis*, i.e., NE↔SW.

Indeed, as assessed by our *literature review*, as stated in *Chapter 3*, *movements* along the *thumb's natural axis*, i.e., NE↔SW, are identified to be *faster* and result in *less physical strain* as *fewer degrees of freedoms* are involved [Trudeau et al., 2012a, Xiong and Muraki, 2014]. In addition, *rolling* achieved *high recognition rates* and was liked by *participants* [Roudaut et al., 2009]. Hence, we decided to utilize the *thumb roll gesture* for *bidirectional interaction designs*.

Pressure Pattern uses *unique pressure patterns* to offer *immediate access* to both directions.

Pressure Pattern depicts the second *switch* in this *section* and offers *direct access* to *both directions*. However, instead of using a *rolling-gesture*, the *mechanism* piggybacks *information* about the *intended direction* by using *unique pressure variations*. Consequently, as illustrated in *figure 4.6 (A)*, *slow pressure exertion* sets the *direction* to *increase* (*yellow area*), while *maximum force*, followed by *slow pressure-release*, results in the *opposite direction* (*blue area*). In this manner, *pattern-changes* are acknowledged, as soon as *force* is *quickly applied*. As a result, *pressure-pattern* features a *natural mapping* that *reduces values* on *pressure-release*. It is important to note that *directions* do not remain *constant*, but rather *turn back* to *increase*, as soon as *pressure* gets *slowly applied*.

Please be informed that the *mechanism* is inspired by *research* conducted by Ramos and Balakrishnan [2007] who presented *PressureMarks* as *novel approach* to encode *additional information* using *unique force variations*. Even though the *authors* only examined their *approach* in context of *selection-action tasks*, like *copy and paste*, it can be easily adapted within our *bidirectional designs* to specify *directions*.

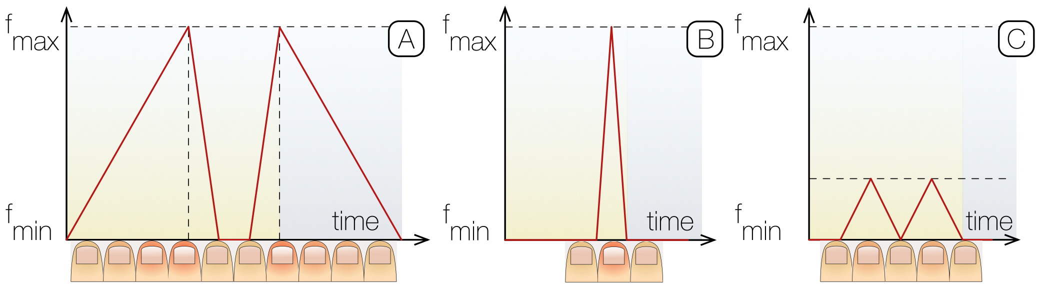


Figure 4.6: Direction Mechanisms: A: *Pressure Pattern*, B: *Maximum Force*, C: *Double Pulse* (yellow area ≙ *increase*, blue area ≙ *decrease*)

Toggles

Maximum Force follows a *simple idea* and is based on the observation that *high pressure targets* are easily accomplished [Heo and Lee, 2012]. A possible *explanation* is given by the fact that *humans* do not have to hit *force levels* precisely, but only have to hit *as strong as possible* to reach the *maximum level*. As a result, we can exploit this ability to *toggle directions*, as illustrated in *figure 4.6 (B)*, where the *direction* remains static, until *maximum-force* is *quickly applied*.

In *maximum force* uses have to quickly apply *maximum force* to toggle directions.

By contrast, ***Double Pulse*** uses the *lower part* of the *force-sensitive range*, since *recent work* in the *area* of *pressure-based interaction* found *low-located targets* to be *more sensitive* than *high located ones* [Ramos et al., 2004]. Thus, we decided to allocate this *area* to a *double-pulse gesture* where *users* have to repetitively exert *little force*, as illustrated in *figure 4.6* (C).

Double pulse requires *repetitive pressure-exertion* at the bottom of the *force-sensitive range*.

Finally, ***Thumb Bob*** depicts the last *toggle*, and relies on the *thumb's movement capabilities*, similar to *thumb roll*. Clearly, the *arrangement* of the *interphalangeal joint* is *well-suited* for *movements* within N↔S (*section 2.2.1*), as confirmed by *research* conducted by Karlson and Bederson [2006b] who explored *thumb ergonomics* within *single-handed device operation*. Hence, we designed the *thumb-bob gesture* to feature a *natural motion* where the *thumb* initially *keeps contact* with the *underlying surface* using its tip, *bobs down* while *increasing* its *contact size*, and immediately comes back up again to *finalize* the *gesture*. Note that *changes* in *contact size* are easily detected [Rekimoto and Schwesig, 2006].

Using *thumb bob*, *users* have to *bob up* and *down* to toggle directions.

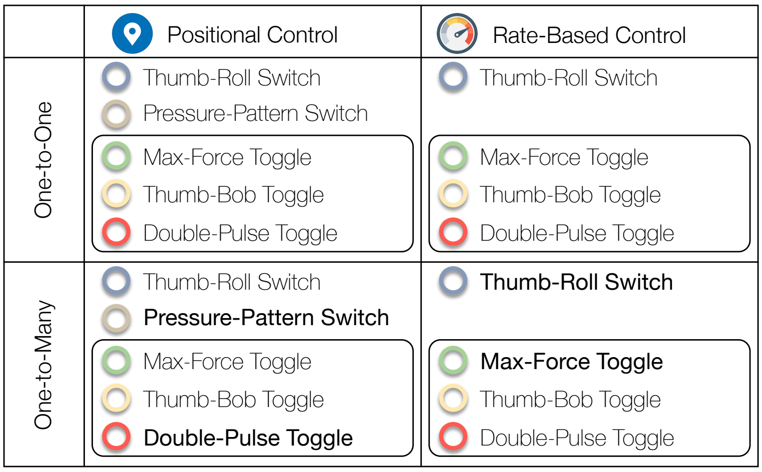


Figure 4.7: Design Space of Bidirectional Force Input: Each combination of *pressure-control mechanism*, *pressure-mapping* and *direction mechanism* yields one *bidirectional design*.

Each combination of the *three essential components* yields one *bidirectional design*.

Having referred to the *three essential components* that are *required* to obtain *bidirectional interaction designs*, we can take a *glimpse* at the *design space* of *bidirectional force input* where each *combination* of *pressure-control mechanism*, *pressure mapping* as well as *direction mechanism*, yields a *bidirectional design* respectively. Please be reminded that *additional components*, like *transfer function* or *selection mechanism*, are not *further investigated*, since they are already *well-studied* in *literature* and can easily be combined with *presented designs*.

4.2 Design Space

The *pressure-pattern switch* is only meant to be used within *positional control*.

Figure 4.7 includes the *design space* of *bidirectional force input*, and illustrates how each of the *three components* is assigned to *one dimension* respectively. In addition, *direction mechanisms* are partitioned in *toggles* and *switches* that are *visually set apart* by *rounded boxes*. Indeed, the *pressure-pattern switch* is only meant to be used in context of *positional control*, since adjustments of the *value-changing speed* seem rather *difficult*, when *force* is *released*. Nevertheless, note that all other *direction mechanisms* can be used interchangeably, yielding *eighteen unique bidirectional designs*.

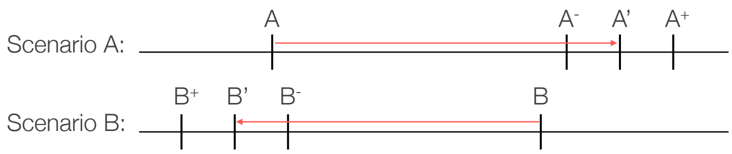


Figure 4.8: Intended Use Case: [Scenario A] value increase from A to A' (*undershoot:* A^-, *overshoot:* A^+), [Scenario B] value decrease from B to B' (*undershoot:* B^-, *overshoot:* B^+).

Before focussing on the *second research question* by conducting an *empirical evaluation* in *Chapter 5*, we *first* draw the *reader's attention* to the *functional concepts* as well as important *implementation details* of presented *bidirectional designs*. In this regard, we refer to *four* out of the overall *eighteen designs*, i.e., to the *designs* indicated in *bold* in *figure 4.7*, since considering them is *sufficient*, to get a *good understanding* about how *bidirectional force input* is accomplished.

Four out of the *eighteen bidirectional designs* are explained as *representative examples*.

4.3 Bidirectional Interaction Designs

Given that the *main objective* of this *thesis* is to come up with an *appropriate solution* to the *bidirectional problem*, we have taken a *glimpse* at the *design space* of *bidirectional force input* and identified *eighteen designs* to allow *value navigation* from a *static location*. Subsequently, the *intended use-case* along with *four representative designs* are discussed.

4.3.1 Intended Use-Case

It is important to realize that all *presented designs* in this *section* share the *same common purpose* and are meant to be used within the following *use-case*: In this regards, *figure 4.8* distinguishes *two common scenarios* in which *values* are *in-* or *decreased* respectively. Indeed, *scenario A* depicts the *first case* where A should be *increased* to A'. However, due to *unintended force variations*, *value manipulations* might end up in *undesired over-*, i.e., A^+ or *under-shoots*, i.e., A^-.

Proposed bidirectional designs share the same *intended use-case*.

Conversely, *scenario B* represents the *opposite case* where B should be *decreased* to B'. Similarly to the *first scenario*, the *intended value* might not be hit *precisely*, resulting in *undesired over-*, i.e., B^+, or *undershoots*, i.e., B^-, respectively. Note that *bidirectional designs* have to *account* for *both scenarios* to allow *bidirectional force input* from a *static location*. Hence, to explain the *functional concepts* of *presented designs*, we utilize the *above-stated use-case* to demonstrate how *bidirectional force input* is accomplished.

Functional concepts of our *designs* are explained according to the *above-stated use-case*.

4.3.2 Functional Concepts

Since explaining all *eighteen designs* would go beyond the *scope* of this *thesis*, *this section* provides an *overview* of the *main concepts* that are *required* to understand *functional concepts* of *presented designs*. Clearly, some of the them feature similarities and hence can be considered jointly. As an example, *designs* containing a *one-to-many mapping* follow *almost* the same *functional concept* as *designs* that are meant to be used within a *single multi-range region*. In addition, *designs* that feature identical *direction mechanisms* and only differ in the utilized *pressure-control mechanism*, are *well-suited* to be considered jointly. As a result, we obtain *four exemplary designs* that are explained in the *following sections*.

The majority of *presented designs* feature *similarities*, and hence can be considered jointly.

One-to-Many Pressure-Pattern Positional Pumping

The first *designs* applies *positional control* and uses the *pressure-pattern switch*, as introduced in *section 4.1.3*, to offer *immediate access* to *both directions*. Please be informed that all of our *designs* include *tactile feedback* to indicate *direction changes*, as well as a *resting threshold* such that *users* can initially *rest* their *thumb* on-screen, without affecting their *current selection*. This decision is *crucial*, since it avoids *unintended changes* that would otherwise cause *usability issues*. As a consequence, as illustrated in *figure 4.9 (A)*, the *value* remains constant, until the *resting threshold* is exceeded (t_0). Whenever this is the case, the *value* increases according to the *amount* of *exerted force*, until the *maximum quantifiable level* of the *force-sensitive resistor* is reached (t_1).

All *bidirectional designs* include *tactile-feedback* as well as a *resting threshold* to avoided *unintended changes*.

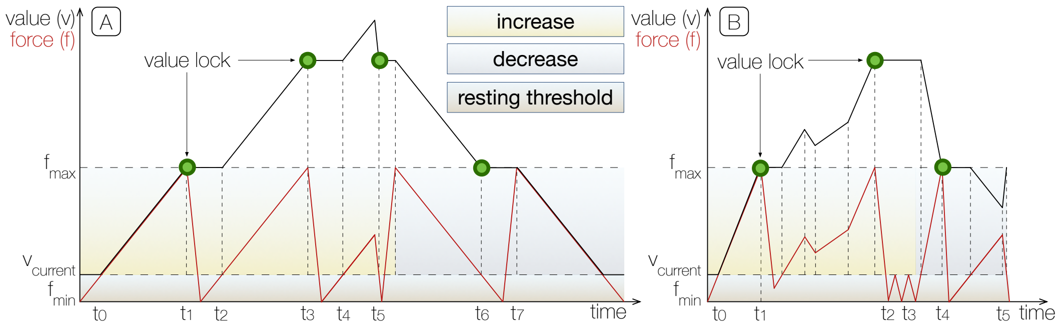


Figure 4.9: Bidirectional Interaction Designs: A: *One-to-Many Pressure-Pattern Positional Pumping*, B: *One-to-Many Double-Pulse Positional Pumping*

To continue within the next *multi-range region*, the concept of *positional pumping*, as stated in *section 4.1.2*, is applied where *values* are locked, as soon as *maximum force* is accomplished. As a result, *users* can return to the *resting threshold* without changing their *current selection* and reapply *force* to acquire *values* that are positioned within the next *multi-range region* (t_2). In addition, *overshoot corrections* within the *current segment* are easily made by *slightly releasing force* until the desired *location* is met (t_4 to t_5). Finally, starting from t_5, *users* can move in the *opposite direction* by quickly applying *maximum force*, followed by *slow pressure release*. Consequently, by repeating this *pattern* the *value* further *decreases* until the *original location* is met. Taken together, *scenario A* and *B* of the *intended use-case*, as stated in *section 4.3.1*, are achieved as follows:

Overshoot-corrections within the *current segment* are easily made using *positional control*.

A: *undershoots*: press stronger
overshoots: release some pressure

B: quickly apply maximum force (*pattern change*), then:
undershoots: release more pressure
overshoots: press stronger

One-to-Many Double-Pulse Positional Pumping

Similar to the *previous technique*, the second *bidirectional design* also makes use of *positional-control*, but rather utilizes the *double-pulse gesture* (*section 4.1.3*) to *toggle directions*. As a result, *users* do not have *immediate access* to *specify directions*, but can rather *toggle* them in alternation.

The *second design* uses a *toggle-* rather than a *switch-* mechanism.

Figure 4.9 (B) visualizes the *functional concept* of this *design* and illustrates how the *double-pulse* is performed. In this manner, the *value* increases until it gets *locked* when *maximum force* is registered (t_1). Next, all *values* within the second *multi-range region* are acquired by *exerting* or *releasing force* respectively (t_1 to t_2). Please be informed that *users* only have to return to their *resting threshold*, and do not have to wait until *pressure* is no longer applied. Indeed, returning to the *user's threshold removes* any *value lock* that is currently set, since it ensures that *pressure variations* do not *modify* the *current selected value*. As a result, *reapplying pressure* acquires *values* within the next *multi-range region*.

Returning back to the *user's resting threshold* removes any *value locks* previously set.

In contrast, *values* located in the *opposite direction*, require to *perform* the *double-pulse gesture*, as stated in *section 4.1.3*. Indeed, the *gesture* is assigned to the *lower part* of the *force-sensitive range*, and hence can be performed without having to *leave* the *resting threshold* (t_2 to t_3). Consequently, as soon as the *gesture* is recognized, the *direction* is set to *decrease*, as indicated by the *blue area* in *figure 4.9*. Please be reminded that the *presented design* still employs *positional-control*, even if the *direction* is currently set to *decrease*. Hence, *releasing force* between t_4 and t_5 increases the *value* to set it back to its *previous location*. Overall, the *intended use-case* (*section 4.3.1*) is accomplished as follows:

The *double pulse gesture* can be performed without leaving the *user's resting threshold*.

A: *undershoots*: press stronger
overshoots: release some pressure

B: toggle directions using the *double-pulse gesture*, then:
undershoots: press stronger
overshoots: release some pressure

One-to-Many Maximum-Force Rate-Based Control

As opposed to the *previous techniques*, the *third design* applies *rate-based control* and lets *users* quickly apply *maximum force* to *toggle directions* (*section 4.1.3*). Please be reminded that *rate-based control* maps *force variations* to the *speed* with which *values* are changing (*section 2.4.2*). As a result, *values* remain *constant* when *force* is no longer applied. To provide the *reader* with a *better understanding* of the *design's concept*, *figure 4.10 (C)* contains an *illustrative example*.

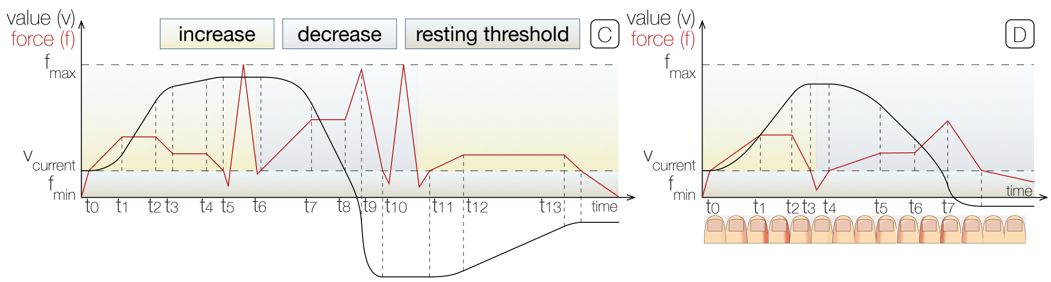


Figure 4.10: Bidirectional Interaction Designs: C: *One-to-Many Maximum-Force Rate-Based Control*, D: *One-to-Many Thumb-Roll Rate-Based Control*

This way, the *value* starts *moving* if more *force* than the *predefined threshold* is observed. Clearly, the *value* increases more quickly from t_0 to t_1, and rises with *constant speed* (t_1 to t_2), until *pressure* is slowly *released* again (t_2 to t_5). However, *moving* in the *opposite direction* requires to return *below* the *predefined threshold* and quickly apply *maximum force* to *toggle directions*. In this manner, the *direction* is changed from t_5 to t_6, resulting in *lower values* as soon as *force* is re-exerted (t_6 to t_9). Finally, the *value* increases until it remains *constant*, as *force* is no longer applied. As a result, the *presented design* involves the *following steps* to enable *bidirectional force input* from a static location:

Movements in the *opposite direction* require *users* to *quickly apply maximum-force*.

A: *undershoots*: press stronger
overshoots: toggle directions; increase pressure

B: toggle directions (*quickly apply maximum force*), then:
undershoots: press stronger
overshoots: toggle directions; increase pressure

One-to-Many Thumb-Roll Rate-Based Control

Finally, the last *bidirectional design* in this *section* combines *rate-based control* with the *thumb-roll switch*, as introduced in *section 4.1.3*. As a result, *users* get *immediate access* to *both directions* by *rolling* their *thumb* either *left* or *right*. Equally important, the *gesture's center* serves as a *resting position* where *values* remain *constant* when *force* is no longer applied. Note that including this *area* is crucial, since it allows *users* to think about their *action* before having an *immediate effect*.

Thumb Roll allows *users* to think about their *action* by rolling to the *resting center*.

Figure 4.10 (D) illustrates the *functional concept* of this *design* and demonstrates how the *value* increases with *rising speed*, as soon as *more force* than the *predefined threshold* is applied (t_0 to t_1). However, please be reminded that instead of *pressing* in the *center*, *force* is exerted while *rolling* in the *respective direction*. Consequently, *force input* is combined with the *rolling gesture* into a *seamless interaction*. Finally, by *rolling left*, the *value decreases* from t_4 to t_7 until *force* is *fully released*. Overall, the *design* enables the *intended use-case* (*section 4.3.1*) using the *following steps*:

Thumb Roll requires users to *exert force* while *rolling* instead of in the *center*.

A: *undershoots*: press stronger while rolling *right*
overshoots: press stronger while rolling *left*

B: *undershoots*: press stronger while rolling *left*
overshoots: press stronger while rolling *right*

Having referred to the *primary concepts* of *four exemplary designs*, the *reader* should be provided with a *better understanding* of how *bidirectional force input* is accomplished. Hence, we have *answered* our *first research question*. Still, the *second research question* requires to evaluate *presented designs* on *actual devices*, to identify the *one* that *performs best* and is *most preferred*. Hence, before conducting an *empirical evaluation* in *Chapter 5*, we refer to *important implementation details*.

Proposed bidirectional designs have provided an *answer* to our *first research question.*

4.4 Implementation

Subsequently an *overview* about the *designs' implementation* is provided. Hence, we briefly refer to the *force-sensing capabilities* of the *apparatus*, explain how *touch events* are handled, and refer to the *main parts* of the *architecture*, i.e., the *design class*, *input controller*, and *direction mechanism*.

4.4.1 Apparatus

To implement the proposed *bidirectional interaction designs*, we decided to utilize an *Apple© iPhone 6s Plus*, since it offers enhanced *force-sensing capabilities* and is *frequently used* in public. Note that we decided for the *larger variant* of the *device* to assess whether our *designs* can overcome *reachability-* and *occlusion-issues* that are typically involved when using *larger phones* within *single-handed device operation*.

An *Apple© iPhone 6s Plus* was used as the *main driver* for the *experiment*.

Please be informed that the device's *form factor* sizes $158.2mm \times 77.9mm \times 7.3mm$ (*height*×*width*×*depth*), and includes an overall *weight* of 192*grams* [Apple©, 2017e]. In addition, a *5.5-inch LED-backlit display* is provided, featuring a resolution of *1920-by-1080-pixel* at 401*ppi* [Apple©, 2017e]. Interestingly, the *display* is built from *multiple layers*, including a *flexible cover glass*, a *transparent capacitive layer*, as well as *strain gauges*, i.e., *force-sensitive resistors* that are located on a 8×12 *grid* underneath the *screen* [Chamary, 2015]. In this manner, the *latter* respond to *physical deformations* and manipulate an *electrical signal* accordingly. As a result, *force-sensing* is enabled by comparing each *strain gauge's signal* to the *local neighborhood* [Chamary, 2015].

Force input is processed by *strain gauges*, i.e., *force-sensitive resistors* that are place underneath the *screen*.

To utilize these *force-sensing capabilities*, we accessed the *force-parameter* as included in the *UITouch-class* contained within *Apple's UIKit framework* to implement *bidirectional designs* using *Swift 3*. This way, as stated in a *detailed evaluation* by Nelson [2015], *force-values* are contained within $[0,400]$ and are divided by 60 to obtain a *maximum possible force* of $400/60 = 6.6666667$ [Nelson, 2015]. However, to let *further calculations* be independent of *absolute values*, we decided to *normalize* the provided *force* using the following *formula*:

$$force = \frac{originalForce}{maximumPossibleForce}$$

Consequently, we obtain $force \in [0,1]$ with 0.15 representing an *average touch* [Nelson, 2015]. Subsequently, the *architecture*, as used for *bidirectional designs*, is stated.

We utilized $force \in [0,1]$ with 0.15 corresponding to an *average touch*.

4.4.2 Architecture

Implementing *bidirectional designs* as proposed in *this chapter* requires an *appropriate architecture* that allows to *reuse* already exiting *components* and offers an *efficient way* to deal with *timeouts*, *interrupts*, and *user interface updates*. Note that *figure 4.11* contains a *simplified version* of the *architecture* and consist of *entities* which are *easily exchangeable* to obtain all *eighteen designs*. Subsequently, *implementation details* regarding the *architecture's major components* are discussed.

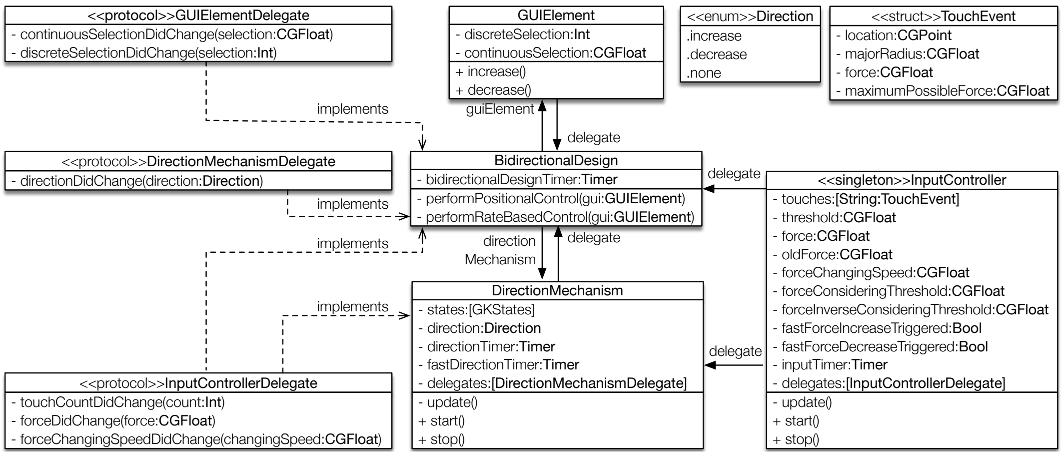


Figure 4.11: Architecture: *BidirectionalDesign:* update-loop, control-mechanism, UI-updates, *InputController:* input-handling, *DirectionMechanism:* direction changes

Bidirectional Design

First, the *bidirectional design class* depicts the *architecture's main component* (*figure 4.11*), and coordinates *input* provided by other *entities* to realize the *desired behavior*. In this regard, it configures the *update-loop*, implements both *pressure-control mechanisms* and signals *upcoming changes* to *user-interface components*. Please be informed that the *update-loop* maintains two different *intervals* in which *changes* are made. This way, *input events* are analyzed every 0.1ms, while *user-interface updates* happen *less frequently*, i.e., every 16ms. This *distinction* is crucial, since it allows to interpret *force-level changes* before deciding about *user-interface adjustments*.

Input events are processed *more often* than *user-interface updates*

In addition, *instances* register *themselves* as *delegates* of the *direction mechanism* and *input controller*, to be notified about *direction-* as well as *force-level updates*. As a result, *bidirectional force input* only requires to *instantiate* a *design* and specify a *graphical element*, the *input* is mapped to.

The *delegation pattern* is used to obtain *loosely coupled components*.

Input Controller

Equally important, the *input controller* depicts an *additional component* of the *architecture*, and is responsible for *handling touch-* as well as *force-level events*. Clearly, as shown in *figure 4.11*, the *component* is realized as a *singleton* and features a *simple interface* using *start()* and *stop()-methods* respectively. Consequently, *bidirectional designs* have the ability to *activate* or *deactivate input-handling* at any time, and only have to conform to the *InputControllerDelegate protocol* to be notified about *upcoming changes*. Nevertheless, even though *input events* are provided by *UIKit* [Apple©, 2017d] by the following *methods*,

```
func touchesBegan(_ touches: Set<UITouch>, with event: UIEvent?)
func touchesMoved(_ touches: Set<UITouch>, with event: UIEvent?)
func touchesEnded(_ touches: Set<UITouch>, with event: UIEvent?)
func touchesCancelled(_ touches: Set<UITouch>, with event: UIEvent?)
```

it is important to realize that they do not get called with a *predefined frequency*, but only get *updated* when *changes* to *touch events* have occurred. As a result, we decided to keep track of each *event's lifecycle* by storing it in a *dictionary* of type [String:TouchEvent], identified by its *memory address*. As a result, *touches* are stored within *touchesBegan(...)*, modified within *touchesMoved(...)*, and discarded whenever either *touchesEnded(...)* or *touchesCancelled(...)* is called. Hence, we can access *touch events' location*, *force* and *radius* at any time using the *above-stated dictionary*.

Touch events were stored in a *dictionary* throughout their *entire lifecycle*.

With this in mind, the *obtained information* is used to calculate *properties* that are required to implement *bidirectional interaction designs*: First, *forceChangingSpeed* $\in [-1,1]$ determines how fast *values* are changing and is calculated according to the *difference* between *current-* and *old-force* respectively. In this regard, *positive values* correspond to an *increase* in *force*, while *negative values* occur during *pressure release*. Second, additional properties, namely *fastForceIncreaseTriggered*, *fastForceDecreaseTriggered* $\in \mathbb{B}$ are *obtained* as follows:

ForceChangingSpeed is calculated to detect a *fast force-increase*.

$$
\begin{aligned}
\mathit{fastForceIncreaseTriggered} &\leftrightarrow (\mathit{forceChangingSpeed} > 0.1) \\
\mathit{fastForceDecreaseTriggered} &\leftrightarrow (\mathit{forceChangingSpeed} < -0.1)
\end{aligned}
$$

Finally, *forceConsideringThreshold* $\in [0,1]$ accounts for the *resting threshold*, as introduced in *section 4.3.2*, and hence removes the *need* to consider it in *further computations*:

$$
\mathit{forceConsideringThreshold} = \frac{max((\mathit{force} - \mathit{threshold}), 0.0)}{1.0 - \mathit{threshold}}
$$

Having referred to the *bidirectional design class* as the *architecture's main component*, as well as to the *input controller*, offering *convenient access* to *input-events*, we finally draw the *reader's attention* to the *direction mechanisms' implementation*, and state how *direction changes* are recognized.

Direction Mechanism

Direction mechanisms were implemented using *state-machines* rather than *decision-trees*.

Implementing *direction mechanisms*, as introduced in *section 4.1.3*, requires to analyze *incoming pressure variations* to decide whether *predefined gestures* have occurred. Unfortunately, checking for *multiple conditions* usually involves large *decision-trees* that are difficult to maintain. Hence, we decided to utilize *state-machines* that reduce the *gestures' complexity*, using *local* decisions in each *individual state*.

Each *state-machine* traverses *multiple states* until the *predefined gesture* is recognized.

Note that *state-machines* have been implemented as *GK-StateMachine*, as provided by *Apple's GameplayKit framework* [Apple©, 2017b] and share a *similar structure* by featuring *ThumbLifted* as initial state (*figure 4.12*). In this manner, each *gesture* is initiated by *resting* the *user's thumb* on-screen and *traverses multiple states* until the *intended gesture* is *successfully detected*. Note that, *lifting* the *user's thumb* returns back to the *initial state*, independent of the *state* that is currently set. As a result, each *state-machine* depicts a *close-loop cycle* and accounts for individual *gesture characteristics*.

Whenever the *direction* has changed, it is confirmed using *tactile feedback*.

In this regard, as illustrated in *figure 4.12* (A), the ***pressure-pattern state-machine*** enters *FastForceIncreaseTriggered*, as soon as $\mathit{forceChangingSpeed} > 0.1$ and $\mathit{force} < 0.5$ are satisfied, indicating that *force* has *quickly increased*. In addition, if *maximum force* is applied, i.e., $\text{force} == 1.0$, the *gesture* is successfully detected, causing the *direction* to be set to *decrease*, followed by a *light bump* using *tactile-feedback*. Otherwise, the *state-machine* reenters *ThumbDetected* if *force* is released below the *predefined threshold*. Equally important, the *direction* remains *static* after the *pattern* has been successfully changed until it is *set back* to *increase* when satisfying the *following conditions* respectively:

$$\mathit{force} < threshold$$
$$\mathit{forceChangingSpeed} > 0.0$$

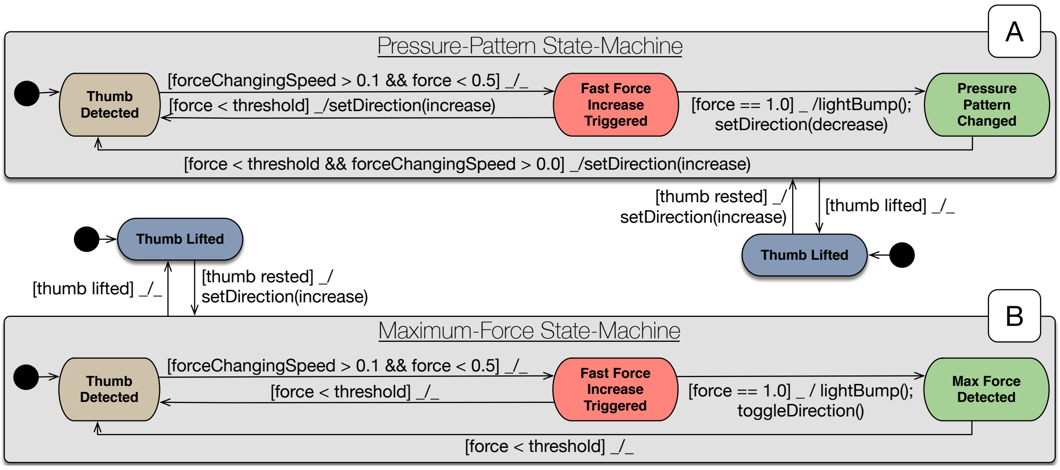


Figure 4.12: State-Machines: A: *Pressure-Pattern-*, B: *Maximum-Force State-Machine*

Moreover the ***maximum-force state-machine*** behaves similar to the *previous one*, but operates as a *toggle* instead of a *switch*. Hence, as illustrated in *figure 4.12* (B), *directions* are only adjusted if the *gesture* has been successfully detected, and is not automatically set back to *increase* if *ThumbDetected* is entered. In this manner, *users* can *alternate directions* by quickly applying *maximum-force*. However, please be informed that *user-interface updates* need to be disabled, as soon as *FastForceIncreaseTriggered* is entered. This is because *traversing* the *force-sensitive range* until *maximum-force* is accomplished, would otherwise result into *visual glitches* that are undesired within *bidirectional interaction designs*. Hence, when *updates* are paused, *users* can quickly apply *maximum force* without affecting *graphical components*.

User-interface updates need to be paused when a *fast-force increase* is detected.

In contrast, the ***double-pulse state-machine*** is *slightly more complex* and contains an *individual state* for *each phase* of the *gesture* respectively (*figure 4.13, A*). Initially, $force \in [0.12, 0.295]$ needs to be satisfied. As a result, the *state-machine* enters *FirstTopReached* and waits for $500ms$ to satisfy $force \in [0.00, 0.11]$. If the condition is met and the *timeout* is not yet exceeded, *FirstLowReached* is entered. Otherwise, the *gesture* is cancelled, and *ThumbDetected* is entered. Analogously, the *state-machine* waits for $500ms$ to enter *SecondTopReached* by satisfying $force \in [0.12, 0.295]$, and *toggles directions* as soon as the *condition* is met. Otherwise, the *state-machine* is reset to *ThumbDetected*.

Double pulse requires that *several states* are visited in a *row* within *predefined time-intervals*.

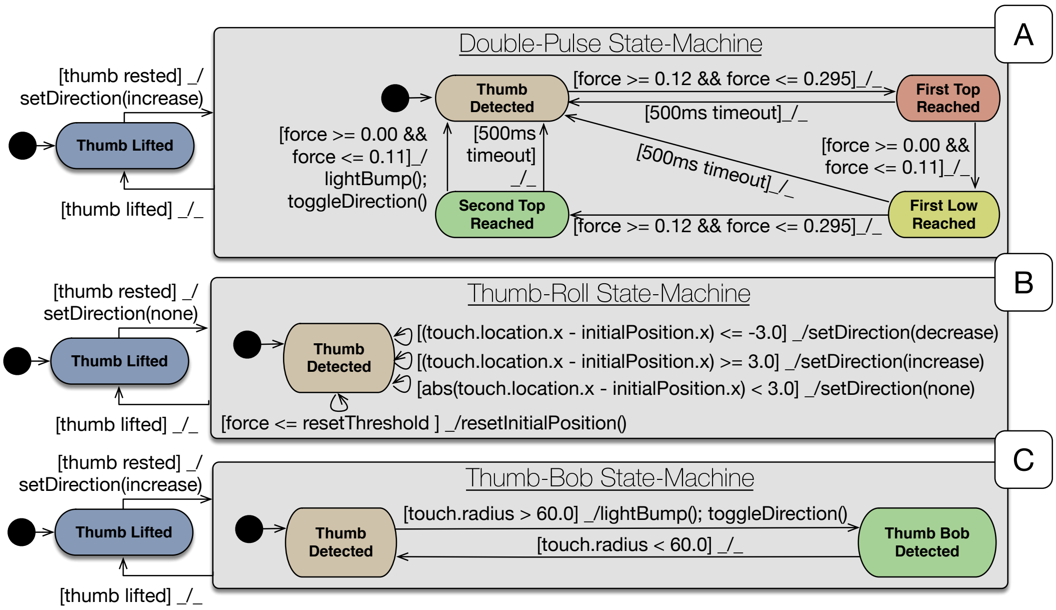


Figure 4.13: State-Machines: A: *Double-Pulse State-Machine*, B: *Thumb-Roll State-Machine*, C: *Thumb-Bob State-Machine*

The *thumb roll state-machine* calculates the *travelled distance* according to a *reference-position*.

Interestingly, the ***thumb-roll state-machine*** (*figure 4.13, B*) is different, since it does not only accounts for *force variations*, but also considers the *thumb's location* to *specify directions*. In this regard, an *initial position* is kept when the *user's thumb* is rested on-screen. Note that this *location* gets updated as long as *force* stays within a *predefined threshold*. As a result, when *rolling left* or *right* the *travelled distance* from the *initial location* increases and is used to *specify directions*.

The *thumb bob state-machine* uses differences in *finger-contact size* to toggle directions.

Finally, the ***thumb-bob state-machine***, as shown in *figure 4.13, C*, follows a rather *simple approach* and only distinguishes between *two additional states*, given the *initial one*. This way, as soon as *touch.radius* > 60.0 is satisfied, the *state-machine* enters *ThumbBobDetected* and triggers a *small bump* to inform *users* that the *direction* has successfully changed.

Having proposed *eighteen bidirectional designs* along with their *functional concepts*, *intended use-case* and *implementation*, we now draw the *reader's attention* to our *second research question*, and conduct an *empirical evaluation* to identify which *combination* of the *three essential components performs best* and is *most preferred*.

Chapter 5

Evaluation: First Study

Given that the main *objective* of this *thesis* is to come up with an *appropriate solution* to the *bidirectional problem*, we carry on with our *systematic procedure* (*Chapter 2*), to provide answers to our *research questions*. While *Chapter 4* has focussed on the *first question*, and identified *three essential components* that are required to enable *bidirectional force input* from a *static location*, it is now *possible* to draw the *reader's attention* to the *second research question*, and conduct an *empirical evaluation*. Hence, the purpose of this *study* is to *investigate* which *combination* of *pressure-control mechanism*, *pressure mapping* and *direction mechanism* performs *best* and is *most preferred* by participants, to identify the *ones* that should rather be excluded from *further considerations*. As a result, we can concentrate on the *designs* that are built from *remaining components* and conduct a *second study* to evaluate their *performance* against a *baseline-condition*.

Main objective of this study: identify which combination of *pressure-mapping*, *direction-* and *pressure-control mechanism* performs best and is *most preferred*.

Subsequently, the *following sections* deal with the *study design*, including *hypotheses*, the utilized *task*, as well as *essential design decisions* made. Equally important, *independent-* as well as *dependent variables*, along with the *experiment's target group*, are stated. Moreover, the *experimental design*, including the number of *resulting conditions*, as well as how *counterbalancing* is achieved, is discussed. Finally, the *chapter* concludes with our *statistical analysis*, and highlights *results* together with *resulting implications* for the *second study*, as stated in *Chapter 6*.

5.1 Hypotheses

Throughout the *study*, we examine the following *hypotheses* (stated in *null form*, i.e., expected to be *rejected*):

H1 Acquiring *targets* using different *direction mechanisms* yields the same *performance* for fixed combinations of *pressure mapping* and *pressure-control mechanism*.

H2 Completing tasks using various *pressure-control mechanisms* results in the *same performance* for fixed combinations of *pressure mapping* and *direction mechanism*.

H3 *User preference* is the same among *direction mechanisms* for fixed combinations of *pressure mapping* and *pressure-control mechanism*.

5.2 Task

A *target-acquisition* and *selection-task* was used to assess *user-preference* and *performance* of *bidirectional designs*.

To evaluate *user preference* and *performance* of *bidirectional designs*, as proposed in *Chapter 4*, we decided to adapt a *target acquisition-* and *selection task*, as used by Ramos et al. [2004], Shi et al. [2008] and Heo and Lee [2012]. In this manner, we ensure *internal validity* by utilizing *research methods* that are *well established*. Consequently, *participants* are asked to perform *sequential target-acquisition* and *selection tasks* as *quickly* and *accurately* as possible by executing the *following steps:*

1. Initially, *users* have to pick up the *device* that is used for *gathering data* and find a *good grip* while operating the smartphone *single-handed* and only using their *thumb*. Indeed, care has to be taken that the *thumb* can easily rest within the *predefined area* (*figure 5.1*) and *exerts force* without *interference*.
2. While resting the *thumb* in the *interaction area*, *users* have to *navigate* to the *intended target* (T), as *quickly* and *accurately* as possible. Note that the *cursor's discrete position* is highlighted in *black* (*figure 5.1*), while its *continuous location* is indicated through a *white line*. Equally important, the *force-range slider* provides *visual feedback* about *force variations*, while the *arrow* next to it, indicates the *current direction* (*figure 5.1*).

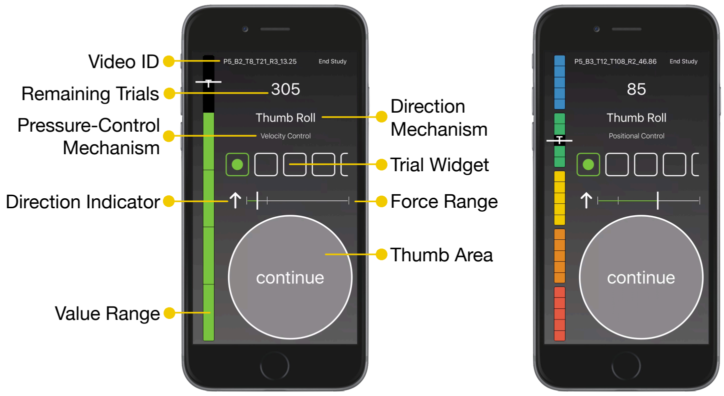


Figure 5.1: iOS Application containing the *target-acquisition* and *selection task*: Left: *one-to-one*, Right: *one-to-many*.

3. Considering that *tasks* consist of *start-* as well as *target position* and are *performed* with *one* of the *eighteen designs*, the intended *direction-* and *pressure-control mechanism* is announced on-screen, while the *cursor* is set to its *starting position*. In addition, the *value-range* is configured to represent a *single* (*figure 5.1, left*) or *multiple regions* (*figure 5.1, right*), depending on the *type* of *pressure mapping* being used. As a result, *users* can perform *bidirectional value manipulations* by specifying the *cursor's movement speed* (*rate-based control*), or *absolute position* (*positional control*).

The intended *direction-* and *pressure-control mechanism* are announced on-screen.

4. Equally important, *participants* can utilize *jump-over points* when using *positional control* combined with *multiple regions* (*section 4.1.2*). This allows them to *fully release pressure* without affecting their *current selection*. As a result, *targets* that are located outside the *current region* can be acquired, since *values* get locked, as soon as *maximum force* is applied.

Jump-over points allow *users* to reach *values*, located outside the *current region*.

5. Finally, when *users* feel confident to have acquired the *intended target*, they can *finalize* their *selection* by using *dwell-time* as *selection mechanism* (*section 2.4.3*). Consequently, *pressure* has to be maintained for $1s$ in case of *positional control*, or for $2.5s$ when *rate-based control* is applied. Nevertheless, in either case the *user's selection* is confirmed using a *short blinking*.

Indeed, the *above-stated steps* are necessary to complete *trials* within the *target acquisition* and *selection task*, as used in this *study*. However, note that several *design decisions* have been made that are justified in the following section.

5.2.1 Task Design Decisions

The *continue-button* is placed to be within *comfortable reach* of the *user's thumb*.

Continue Button *First*, we decided to position a *continue-button* (*figure 5.1*) underneath the *user's thumb*, immediately after a *task* is completed. Hence, *participants* do not need *significant changes* in *hand posture*, but rather proceed to the *next trial* by simply *tapping* a button. Note that this *design decision* is motivated by our aim to enable *bidirectional force input* from a *static location*, and also ensures that *force* is *completely released*, before the *next trial* is encountered.

Trials can be undone using the *trial widget*.

Trial Widget Equally important, *users* might experience *accidental mistakes* due to the *novelty* of the *presented designs*. To tackle this *issue*, we decided to include the *trial widget*, as illustrated in *figure 5.1*, that offers the *opportunity* to repeat *tasks* that are already completed. However, it is important to realize that *any*, rather than only the *target segment* can be selected. This allows to identify *weaknesses* of *bidirectional designs*, since *errors* are registered as soon as they are made.

Feedback about the following is provided:

- *cursor location*
- *exerted force*
- *discrete selection*
- *direction changes*

Feedback In addition, we decided to include *visual feedback* in various ways. *First*, *continuous feedback* about the *cursor's current location*, as well as the amount of *exerted force* is provided. *Second*, information about the *user's discrete selection* is offered at any time by highlighting the *current selected segment* in *black*. Finally, an *arrow* is shown to notify *users* about ongoing *direction changes* that are also accompanied by *light bumps* using *tactile feedback*. Note that these decisions are justified by *research* conducted by Wilson et al., who have referred to the importance of *continuous feedback* in context of *pressure-based interaction* [Wilson et al., 2010]

Selection Mechanism Even though we do not investigate the impact of *selection mechanisms*, since our aim is to find appropriate *solutions* to the *bidirectional problem*, including a *method* to *select* is crucial to conduct an *empirical evaluation*.

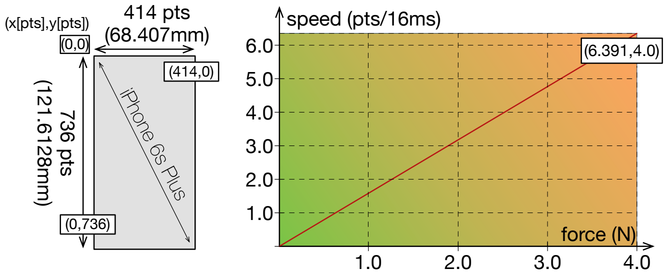


Figure 5.2: left: *screen dimensions* of *iPhone 6s Plus*, right: *transfer-function* for *rate-based control* by Wilson et al. [2011].

Hence, we decided to utilize *dwell-time* as *selection modality*, since it is found to offer *reliable result*, despite causing major delays [Cechanowicz et al., 2007, Ramos et al., 2004]. Consequently, we omit the impact of different *selection mechanisms* and allow *consistent comparisons* among *bidirectional designs*. Note that our decision, to utilize *shorter durations* in case of *positional control* ($1s$ vs. $2.5s$), is motivated by *research* conducted by Heo and Lee [2012], who found that *maintaining force* is difficult over long periods of time. Nevertheless, *bidirectional designs* can later be combined with various *selection mechanisms* to obtain *faster selections*.

Dwell-time served as selection mechanism throughout the *study*.

Transfer Function Evaluating *transfer-functions* is beyond the scope of this *thesis*. Hence, we looked at *recent work* to identify *functions* that are commonly used. Regarding *positional control*, Stewart et al. [2010] identified a *linear transfer-function* to work best if the *sensor's input* is linearized using an *op-amp current-to-voltage circuit*. Consequently, we assign *maximum force* to the *highest value*, and linearize *remaining levels* accordingly. Similarly, in case of *rate-based control*, we adapt the *transfer function* of Wilson et al. [2011], yielding $66mm/s$ when *maximum force* is applied. Still, we decided to *double* the *speed* of Wilson et al., since *long-distance targets* felt *too slow* during *initial testing*. Hence, we obtain a *maximum speed* of $2*6.39090622pts/16ms$, considering the *device's screen size* (*figure 5.2*) and the *rule of three*:

For both *positional-* as well as *rate-based control linear transfer functions* are used.

$$\begin{aligned} 1\,s &\mathrel{\widehat{=}} 66\,mm & 121.6128\,mm &\mathrel{\widehat{=}} 736\,pts \\ \Rightarrow 0.016\,s &\mathrel{\widehat{=}} 1.056\,mm & \Rightarrow 1.056\,mm &\mathrel{\widehat{=}} 6.39090622\,pts \end{aligned}$$

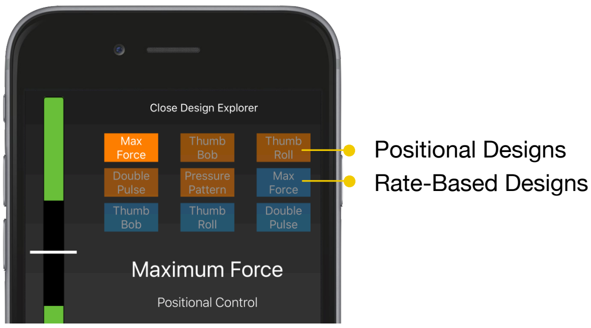


Figure 5.3: Design Explorer: explore *bidirectional designs* using *positional-* (*orange*), as well as *rate-based control* (*blue*).

Breaks are offered whenever needed.

Setting Referring to the *overall surrounding*, *participants* are seated on a *regular chair*, measuring $44.0cm \times 43.5cm \times 55.0cm$ ($length \times width \times height$) in size. Note that considerable care had to be taken that *participants* do not *rest* or *stabilize* their *arm*, while performing the task *single-handed* and only using their *thumb*. Hence, we decided to choose a *chair* without *arm-rest* that offers *great flexibility* without *interference*. In addition, we decided to offer *breaks* whenever needed to give *participants* a *chance* to recover such that they do not become *fatigue* while *performing* the *task*.

The *design explorer* allows *participants* to familiarize themselves with *proposed bidirectional designs*.

Design Explorer Finally, we decided to include a *design explorer* that allows *users* to try out all *direction mechanisms* that are used in *above-stated designs*. In this manner, *users* have the opportunity to *familiarize* themselves with the *available options* to specify *directions*. In addition, *participants* can develop a *feeling* for how much *force* is required to navigate to the *intended location*. Consequently, we try to minimize any *adverse effects* caused by *potential learning effects* such that they do not *confound* our *results*. Nevertheless, please be aware that the *design explorer* only contains a *restricted subset* of all *eighteen designs*, since it only allows to *navigate* within a single *multi-range region*. Indeed, this *decision* is justified by an *initial observation* that considering all *eighteen designs* in the beginning, is too *mentally demanding*, and would have led to *confusion*. Hence, we have to ensure that *tasks* containing a *one-to-one* mapping are completed, before having to navigate within *multiple regions*.

Having referred to the *study's task* along with *justifications* of important *design decisions* made, the *reader* should be provided with a *better understanding* of the *target acquisition* and *selection task*, as used in this *study*. Hence, the following *sections* deals with the *resulting design*, including *independent-* as well as *dependent variables* that are used to evaluate *bidirectional designs*.

5.3 Design

5.3.1 Independent Variables (Factors)

Throughout the *study* we control the *following* conditions:

Technique The main *factor* of this *study* is *technique*, i.e., one of the *proposed interaction designs* to enable *bidirectional force input* from a *static location*. Indeed, when controlling *technique* we also implicitly determine the *pressure mapping*, *direction-* as well as *pressure-control mechanisms* that is used to navigate to the *desired location*. Hence, *technique* is easily controlled by announcing the *components' name* on-screen and configuring the *value-range* to fit the *pressure mapping* being used. Subsequently, corresponding *levels* are stated:

When controlling *technique*, we implicitly control the *pressure mapping*, *direction-* as well as *pressure-control mechanisms*.

ID	Technique
T_1	One-to-one *max-force* positional navigation
T_2	One-to-one *thumb-bob* positional navigation
T_3	One-to-one *thumb-roll* positional navigation
T_4	One-to-one *double-pulse* positional navigation
T_5	One-to-one *pressure-pattern* positional navigation
T_6	One-to-one *max-force* rate-based control
T_7	One-to-one *thumb-bob* rate-based control
T_8	One-to-one *thumb-roll* rate-based control
T_9	One-to-one *double-pulse* rate-based control
T_{10}	One-to-many *max-force* positional pumping
T_{11}	One-to-many *thumb-bob* positional pumping
T_{12}	One-to-many *thumb-roll* positional pumping
T_{13}	One-to-many *double-pulse* positional pumping
T_{14}	One-to-many *pressure-pattern* positional pumping
T_{15}	One-to-many *max-force* rate-based control
T_{16}	One-to-many *thumb-bob* rate-based control
T_{17}	One-to-many *thumb-roll* rate-based control
T_{18}	One-to-many *double-pulse* rate-based control

Table 5.1: Levels of *Technique* (Study 1).

Note that *above-stated techniques* are partitioned into *four different blocks* depending on their *characteristics:*

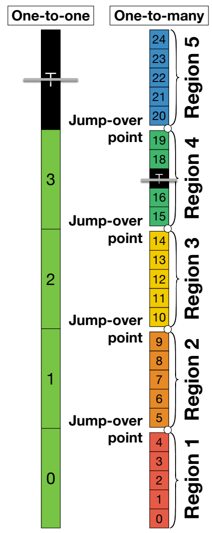
$$block_1 := \{T_1, \ldots, T_5\} \qquad block_3 := \{T_{10}, \ldots, T_{14}\}$$
$$block_2 := \{T_6, \ldots, T_9\} \qquad block_4 := \{T_{15}, \ldots, T_{18}\}$$

In addition to *regular trials*, *test trials* are included to *familiarize* with the *target-acquisition* and *selection-task*.

Task *Task* depicts the *second factor* in this *study* and is defined as a *tuple*, containing *discrete start-* and *target-position* respectively. Note that *tasks* are defined for both *pressure mappings*, and are categorized depending on whether the *target* is positioned *above* or *below* the *original location*. In addition, please be aware that *tasks* are chosen to represent the *value-range* as good as possible. Consequently, *repetitions* can be included to obtain *more reliable* results. Equally important, *test trials* are included to become *familiar* with the *intended design*. Subsequently, *levels* are stated:

$$task \in \text{Tasks}^{test}_{\text{one-to-one}} \cup \text{Tasks}^{test}_{\text{one-to-many}} \cup \text{Tasks}^{above}_{\text{one-to-one}} \cup \text{Tasks}^{below}_{\text{one-to-one}} \cup \text{Tasks}^{above}_{\text{one-to-many}} \cup \text{Tasks}^{below}_{\text{one-to-many}}$$

Set	Tasks
$\text{Tasks}^{test}_{\text{one-to-one}}$	{(1,3), (3,1)}
$\text{Tasks}^{test}_{\text{one-to-many}}$	{(5,23), (23,5)}
$\text{Tasks}^{above}_{\text{one-to-one}}$	{(0,2), (1,4), (0,4)}
$\text{Tasks}^{below}_{\text{one-to-one}}$	{(2,0), (4,1), (4,0)}
$\text{Tasks}^{above}_{\text{one-to-many}}$	{(7,13), (3,17), (2,24)}
$\text{Tasks}^{below}_{\text{one-to-many}}$	{(13,7), (17,3), (24,2)}



5.3.2 Dependent Variables (Measures)

By controlling *Task* and *Technique* we ensure that *participants* perform the *target-acquisition* and *selection task* under different conditions. However, drawing *conclusions* about which combination of *pressure mapping*, *direction-* as well as *pressure-control mechanism performs best* and is *most preferred*, requires *appropriate measures* to assess *differences* of proposed *bidirectional designs*. Hence, *values* of the following *dependent variables* are calculated:

- **Task Completion Time [*seconds*]** depicts the *first dependent variable* and is defined by the *total time* that is required to *acquire* and *pick* a *segment* using *dwell-time* as *selection-mechanism*. Consequently, a *stopwatch* is used to keep *track* of the *elapsed time* until a *selection* is made. This way, the *stopwatch* is started, as soon as *participants* rest their *thumb* on-screen and gets stopped whenever a *segment* is confirmed using a *short blinking*. Indeed, drawing *comparisons* between *bidirectional designs* that are using different *pressure-control mechanisms*, requires to *subtract* the associated *dwell-time*, i.e., 1.0s vs. 2.5s, to obtain *fair results*.

Since *pressure-control mechanisms* use *different dwell-times*, *durations* need to be subtracted to obtain *fair results*.

- Moreover, **Target Accuracy [*true*, *false*]** represents the *second measure* and indicates whether the *intended* or any other *segment* is selected. Thus, *target-accuracy* allows to calculate the *number of times* an error occurred, i.e., how often a *wrong segment* is chosen.

- Similarly, **Number of Crossings [count]** provides information about *user's controllability* while performing the *task*. Consequently, this *measure* depicts how often users *over-* or *undershoot* the *intended target* before completing their *choice*.

Crossings indicate how *well* participants can control *bidirectional designs*.

- Finally, **User-Preference [*7-point likert-scale*]** provides *insights* about *users' personal experience* when completing the *task* using *one* of the *eighteen designs*. In this regard, while *previous measures* have drawn attention to *quantitative data*, *user-preference* focusses on *qualitative data* and represents the *last measure* that is used in this *study*.

User-preference allows to assess *qualitative data*.

Please be informed that *above-stated measures* are adapted from *previous studies*, as conducted by Ramos et al. [2004], Shi et al. [2008] and Heo and Lee [2012]. This way, the *combination* of *task-completion-time* and *target-accuracy* allows *judgements* about *users' overall success-rate*, whereas *number of crossings* provides information about the achievable *level-of-control*, when using *proposed bidirectional designs*. Having referred to the *independent-* as well as *dependent variables* of this *study*, we can finally draw the *reader's attention* to the *resulting experimental design*, including the *number of conditions*, and how *counterbalancing* is achieved.

Task-completion-time and target-accuracy provide information about *user's success-rate*.

5.3.3 Experimental Design

The study uses a *within-subject design*.

Turning to the *experimental design*, we decided to choose a *within-subject design*, where each *participant* is presented with all of the *conditions*. As a result, we mitigate *potential biases* due to *individual differences* and only require a *limited number* of participants. However, it is important to realize that choosing a *within-* rather a *between-subject design* raises additional challenges that cannot be neglected. Hence, we have to account for *carry-over effects*, like *learning-effects* and also have to consider that *participants* might become *fatigue* while performing the *target-acquisition* and *selection task*, as used in this *study*. Note that these *issues* can be alleviated by *counterbalancing*, as well as *sufficient breaks* to *recover*.

Pressure mapping and pressure-control mechanism remain static within *blocks*.

Nevertheless, even though a *total randomization* of *conditions* would provide the *necessary balance*, as requested *above*, it also requires *participants* to *alternate* between *designs* that are *fundamentally different*. As an example, we assume that requesting *participants* to frequently *switch* among different *mappings* or *control mechanisms* would cause *confusion* that would inevitably confounds our *results*. Thus, we decided to keep these *characteristics* constant within *blocks*, and only *randomize* within *designs* that differentiate in the utilized *direction mechanism*.

Care has been taken that *one-to-one mappings* are always encountered before *one-to-many mappings*.

Moreover, since we assume that *tasks* that are featuring a *one-to-many mapping* are *more difficult* to perform, we decided to let *participants* perform $block_1$ always before $block_3$ and $block_2$ always before $block_4$ respectively. Still, the *choice* whether *users* start with $block_1$ or $block_2$ is *equally distributed* among *participants* to minimize the *impact* of *ordering effects*. That's why we aim for an *even number* of *participants*.

Subsequently, we remind the *reader* of the *study conditions*:

- 18 *techniques* ($T_1, \ldots, T_{18}$), split into *4 blocks* (*sec. 5.3.1*)
- 2 *tasks* $\in \text{Task}^{test}_{\text{one-to-one}} \cup \text{Task}^{test}_{\text{one-to-many}}$
- 3 *tasks* $\in \text{Task}^{above}_{\text{one-to-one}} \cup \text{Task}^{above}_{\text{one-to-many}}$
- 3 *tasks* $\in \text{Task}^{below}_{\text{one-to-one}} \cup \text{Task}^{below}_{\text{one-to-many}}$
- 3 *repetitions* for each condition

Consequently, we obtain a $18\times(3+3)$ *factorial design* where *each participant* performs $18\times2+18\times(3+3)\times3=360$ *trials*, yielding a *total duration* of $(360\times10s)/60s=60min$ per participant (assuming ≈10s per *trial*). Finally, we conclude the *study design* by referring to the *target group* the *evaluation* is meant for.

The study took ≈ 60 *min* per participant.

5.3.4 Participants

Given that the main *objective* of this *evaluation* is to gain *initial insights* about which combination of *pressure-mapping*, *direction-* as well as *pressure-control mechanism performs best* and is *most preferred*, ten *users* were recruited to participate in the *study*. In this manner, we aimed for *sufficient data* to identify *combinations* that are *most promising*, and which should rather be omitted from *further considerations*.

Ten participants took part in the study.

Equally important, a great deal of *attention* had to be paid to ensure that *participants* neither suffer from *hand injuries* nor have restricted *motor capabilities*, to minimize the impact of *extraneous variables*. In addition, we decided to only focus on *right-handed people* and aimed for an almost *uniform distribution* of *gender*, i.e., *four female* vs. *six male*, to yield *better comparisons*. Finally, participants were aged between 24 and 58 ($M=30.0, SD=10.033$) and have already been *familiar* with *multi-touch interaction*.

Care had been taken that all *participants* are *right-handed*.

Having stated the *experimental design* along with the *desired target group*, the *following section* discusses how *measurements* are handled, before analyzing *results* in *section 5.5*.

5.4 Data Management

Conducting an *empirical evaluation* of *bidirectional designs*, as proposed in *Chapter 4*, does not only require *appropriate measures*, but also demands for a *proper way* to handle the *data*, to make it accessible for *later evaluations*. Consequently, we decided to utilize *comma separated files* (csv-files) that store

Columns containing **context information**:
participantID $\in \mathbb{N}$
processingIndex $\in \mathbb{N}$
trialID $\in \mathbb{N}$
repetition $\in \mathbb{N}$
type $\in$ {test, regular}
timestamp (yyyy-MM-dd-HH:mm:ss:)
Columns containing *independent variables* (**IVs**):
technique $\in \{T_1, \ldots, T_{18}\}$
- directionMechanism $\in$ {maxForce, thumbBob, thumbRoll, doublePulse, pressurePattern}
- pressureMapping $\in$ {one-to-one, one-to-many}
- pressureControlMechanism $\in$ {positional control, rate-based control}
- block $\in \{block_1, \ldots, block_4\}$
task $\in \text{Tasks}^{\text{test/above/below}}_{\text{one-to-one}} \cup \text{Tasks}^{\text{test/above/below}}_{\text{one-to-many}}$
Columns containing *dependent variables* (**DVs**):
selectedValue $\in \{0, \ldots, 24\}$
targetSelectionTime $\in \mathbb{R}_{\geq 0}$ [*s*]
successfulSelection $\in \mathbb{B}$ (*false* $\widehat{=}$ *error*)
numberOfCrossings $\in \mathbb{N}$
[optional] Columns containing *input data* (**INPUT**):
elapsedTimeSinceStudyStart $\in \mathbb{R}_{\geq 0}$ [*s*]
elapsedTimeSinceTrialStart $\in \mathbb{R}_{\geq 0}$ [*s*]
touchX, touchY $\in \mathbb{N}$ [*pts*]
touchRadius $\in \mathbb{R}_{\geq 0}$ [*pts*]
force $\in [0.0, 6.67]$
continuousSelection $\in \mathbb{N}$

Table 5.2: Data Format (Study 1): Column names including associated *Types*.

Measurements are stored in *csv-files* and are exported via Apple's AirDrop.

measurements for each *participant* respectively. In this manner, *individual files* can later be combined into a single *csv-file*, containing all *measurements* categorized by the *participant ID*. Note that *csv-files* are generated using the *csv-export library*, as offered by Cilia [2017], and are stored on *disk* as soon as a *study* is completed. As a result, the *principal investigator* can access *individual files* and export them using *Apple's AirDrop functionality* [Apple©, 2017a].

Columns are partitioned into *context information*, *independent-* and *dependent variables*.

Turning to the *data format* with which *measurements* are stored, *columns* within *csv-files* are structured according to three *major types*, namely *context information*, *independent-* and *dependent variables* (*table 5.2*). First, *context information* includes the *participant-* and *trial-ID*, the *processing index* with a *dedicated timestamp*, along with the *task type*, indicating whether a *regular-* or *test trial* is encountered (*table 5.2*). Please be informed that the *processing index* denotes the *order* in which *tasks* are accomplished. Second, *independent variables* include the *technique*, along with corresponding *components*, like *pressure mapping*, *direction-* and *pressure-control mechanism*, as well as the *block*, the *design* is contained in (*table 5.2*). Likewise, *tasks* with associated *start-* and *target-locations* are considered. Finally, *measurements* of the *dependent variables* are stored, including *task-completion time*, resulting *number of crossings*, as well as a *boolean value*, indicating whether the *task* was *successful*.

Equally important, as illustrated in *table 5.2*, an *optional type* is available that contains *continuous input data* and hence allows to examine, how *bidirectional designs* are applied to *navigate* to an intended location. This way, we distinguish between *two separate files*, i.e., *results.csv* as well as *input.csv*, where the *later* also includes the *user's touch location* with associated *radius*, *force variations*, as well as the *current selected value* that are logged every $16ms$.

In addition to *responses*, *continuous input data* is logged throughout the study.

Having described the *study design* as well as how *measurements* are stored for *later evaluations*, we finally draw the *reader's attention* to our *statistical analysis*, along with resulting *implications* for the *second study*, as stated in *Chapter 6*.

5.5 Study Results

Please be reminded that the *purpose* of the *first study* is to identify which *combination* of the *three essential components*, as required for *bidirectional force input* from a *static location* (*section 4.1*), *performs best* and is *most preferred*. In this regard, *findings* allow us to omit *combinations* that should be *avoided*, and only consider *remaining designs* by evaluating their *performance* against a *baseline-condition* (*Chapter 6*). Note that *data* has been collected from *ten participants* (*4 female, 6 male*, all *right-handed*), according to *three responses*, namely *task-completion time*, *target-accuracy* and *number of crossings*.

Subsequently, the *statistical analysis*, along with its *overall procedure* is stated. In this manner, *findings* of *two major tests* are discussed. Finally, this *section* concludes with *findings* regarding *qualitative data*, and *concludes* with *resulting implications* for the *second study*, as stated in *Chapter 6*.

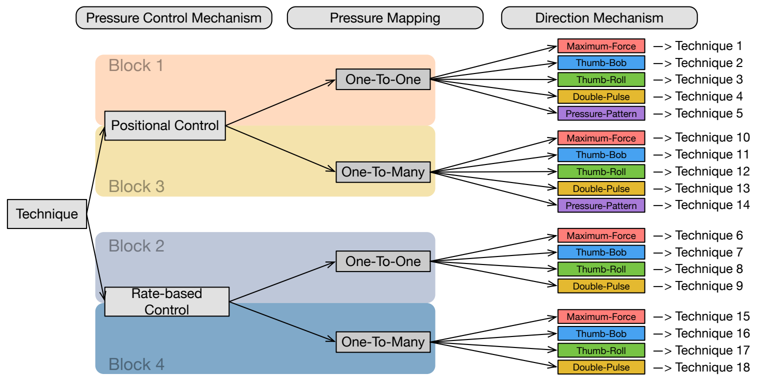


Figure 5.4: Bidirectional Design Overview: Every *combination* of *pressure mapping*, *direction-* and *pressure-control mechanism* yields one *bidirectional interaction design* (*technique*).

5.5.1 Procedure

Target accuracy was omitted from the *statistical analysis*, since *error counts* were low.

Returning back to our *hypotheses*, as stated in the beginning of this *chapter* (*section 5.1*), the *aim* of this *analysis* is to determine whether $H_1, \ldots, H_3$ should be *accepted* or *rejected*. Hence, we focus on H_1 and H_2 by assessing differences in *performance* in terms of *completion time*, *number of crossings* and *error count*. Note that the *latter* is derived from *measurements* regarding *target-accuracy*, obtained in this *study*. However, *error counts* have been identified to be *low*, i.e., most of the times *participants* have selected the *proper target*. Thus, we decided to evaluate *performance* only in terms of *completion time* and *number of crossings*. Finally, H_3 is analyzed using *qualitative data*, derived from a *questionnaire* that was completed throughout the *study*.

The *first test* examines which *direction mechanism* is *best-suited* for a *given block*.

Figure 5.4 reminds the *reader* of how *bidirectional interaction designs* are combined from the *three essential components*, introduced in *Chapter 4*. Hence, to determine *well-suited combinations* of *pressure-mapping*, *direction-* and *pressure-control mechanism*, we identify *three possible tests*, as stated in *table 5.3*. In this manner, the *first test* focusses on H_1 and draws *comparisons* among *direction mechanisms*. Consequently, the *combination* of *pressure mapping*, as well as *control mechanism* remains *static* within each of the *blocks*, to identify which *directions mechanism* performs *best* in terms of *completion time*

Test	Fixed	Comparisons among	Aim/Remark
T_1	*pressure mapping, pressure-control mechanism*	*direction mechanisms*	*Aim:* Investigate which *direction mechanism* performs best for fixed combination of *pressure mapping* and *control mechanism*, i.e., for a *fixed block* (**H1**).
T_2	*pressure mapping, direction mechanism*	*pressure-control mechanisms*	*Aim:* Examine which *pressure-control mechanism* performs best for fixed combinations of *pressure mapping* and *direction mechanism* (**H2**).
~~T_3~~	*pressure-control mechanism, direction mechanism*	*pressure mappings*	*Remark:* Test 3 is not a *reasonable choice*, since comparisons among *pressure mappings* are *unfair*.

Table 5.3: Possible Tests according to the *Three Essential Components*.

and *number of crossings* (*table 5.3*). In contrast, the *second test* concentrates on H_2, and compares among *pressure-control mechanisms* to determine the *one* that *performs best* for *fixed combinations* of *pressure mapping* and *direction mechanism*. Finally, even though the *third test* would be a *logical consequence* with respect to the *previous ones* (*table 5.3*), it is not a *reasonable choice*, since *comparisons* among *pressure mappings* are *unfair*, due to differences in *size*. As a result, we obtain *two statistical tests* to assess H_1 and H_2 respectively.

The *third test* is not a reasonable choice, since *comparisons* among *pressure mappings* are unfair.

Unfortunately, *number of crossings* represents *count data*, and hence is not suited for an *analysis of variance* (short: *anova*). Hence, we decided to use a *nonparametric test*, i.e., an *aligned rank transform* (ART), as proposed by Wobbrock et al. [2011]. Note that the *analysis* is conducted in *R* using the ARTool, as provided by Kay [2017]. Equally important, in case of *completion time* an *anova* may apply if *normality* is ensured beforehand. This is because *completion time* depicts *continuous data* and *factors* are nominal. Thus, we subsequently refer to the necessary *normality test* regarding *task-completion time*, and present *findings* for *each test* respectively.

5.5.2 Normality Test

Applying an *analysis of variance* not only requires that the *response* is measured on a *continuous scale*, but also that the *distribution* is approximately *normal* [Adam and Lund, 2017]. Unfortunately, the *assumption* of *normality* for *completion time* is not met, as assessed by *visual inspection* of *normal Q-Q plots* and *histograms*, as illustrated in *figure 5.5, left*.

Task-completion time was not *normally distributed*.

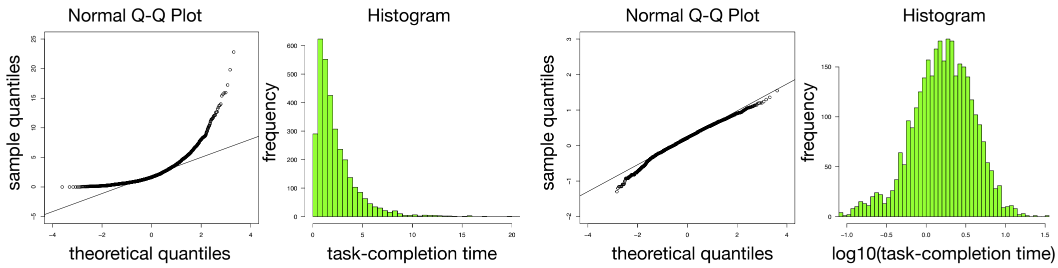


Figure 5.5: Normality Test (*Task-Completion Time*): *left:* assumption of *normality* is violated (*positive skew*), *right:* assumption of *normality* is met for log_{10}-*transformed data*.

Task-completion time was log_{10}-*transformed* before applying an *analysis of variance*.

Hence, we decided to apply a log_{10}-*transformation* to obtain log_{10}(*task-completion time*), since the *data* showed a *positive skew*. Fortunately, inspecting the *transformed data*, the *assumption* of *normality* is met (*figure 5.5, right*). As a result, we applied the *statistical analysis* to the *transformed data* and obtained *results*, as discussed in the *following sections*.

5.5.3 First Test [T1]

Please be reminded that the *purpose* of the *first test* is to determine which *direction mechanism* performed *best* for fixed *combinations* of *pressure mapping* as well as *pressure-control mechanism*, i.e, for *each* of the *four* different *blocks*, as identified in figure 5.4. This way, *results* regarding *task-completion time* and *number of crossings* are stated along with a *discussion* to assess H_1. Note that the *responses' means* are illustrated in *figure 5.6*, along with *error-bars* representing 95% *confidence intervals* (CIs). In addition, *interconnected lines* between *bars* indicate *conditions* that are *significantly different*.

Task-Completion Time

The REML-method was used to account for *possible learning effects*.

To access *differences* between *direction mechanisms* in terms of *completion time*, a *mixed-effect model* analysis combined with the REML-method, i.e., *Restricted Maximum Likelihood*, was used to account for possible *learning effects* and consider *participant* as a *random-factor* [Wobbrock, 2017].

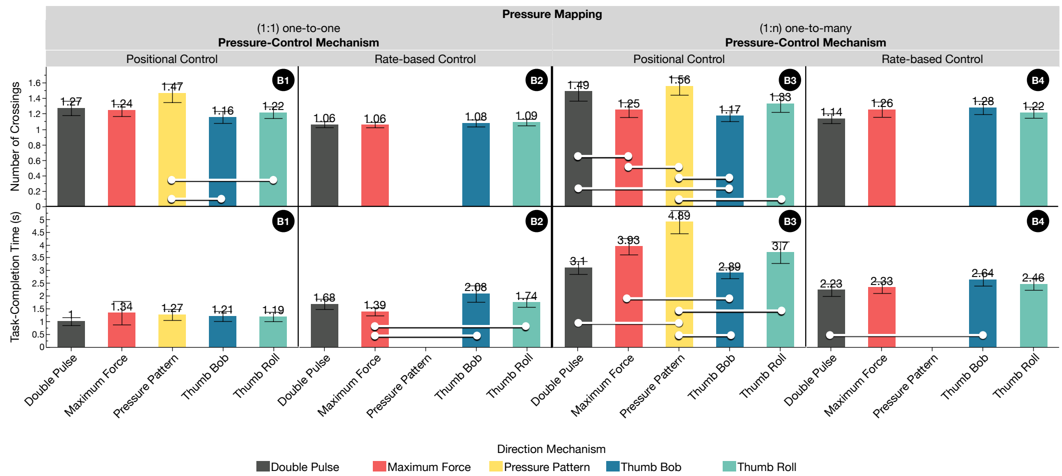


Figure 5.6: Bar-charts representing *means* in **T1**: bottom: *task-completion time*, top: *number of crossings* (error bars: 95% CIs, connection lines: *significant difference*).

Consequently, it did not matter who in particular took part in the *study*, but it had to be ensured that *participants* conform to *requirements*, as stated in *section 5.3.4*, like being *right-handed* and not suffering from *serious hand diseases*. Note that *significance* was accepted at $\alpha = 0.05/4 = 0.0125$, since *four comparisons* are made.

Significance was accepted at $\alpha = 0.0125$.

Subsequently, *results* for *blocks* (**B1**,...,**B4**) are stated:

B1 The *analysis* did not show a *significant main effect* of *direction mechanism*, when using *positional control* combined with a *one-to-one mapping*, $F(4,886) = 1.41, n.s.$. Still, as illustrated in *figure 5.6*, *double pulse* performed *fastest* ($M = 1.0, SD = 1.05$), and was $0.34s$ *quicker* than the *slowest condition*, i.e., *maximum force* ($M = 1.34, SD = 3.11$). However, *choosing* one *mechanism* over the *other* did not lead to *significant changes*.

Direction mechanism had no *significant main effect* on *completion time* within the *first block*.

B2 In contrast, when *rate-based control* is applied, *direction mechanism* showed a *significant main effect* on *completion time*, $F(3,707) = 11.26, p < .00001$. *Post hoc pairwise comparisons* with *Bonferroni correction* revealed that *maximum force* ($M = 1.39, SD = 1.08$) was *significantly faster* than *thumb bob* ($M = 2.08, SD = 2.18$), $t(707) = 5.354, p < .0001$ and *thumb roll* ($M = 1.74, SD = 1.19$), $t(707) = 4.613, p < .0001$. However, note that all other *differences* were not *significant*.

Direction mechanism had a *significant main effect* on *completion time* within B2.

Block	Direction Mechanism	Mean	Median	SD	SE	Block	Direction Mechanism	Mean	Median	SD	SE
B1	Double Pulse	1.00	0.73	1.05	0.08	B3	Thumb Bob	2.89	2.54	1.45	0.11
B1	Thumb Roll	1.19	0.85	1.25	0.09	B3	Double Pulse	3.10	2.73	1.71	0.13
B1	Thumb Bob	1.21	0.79	1.33	0.10	B3	Thumb Roll	3.70	3.11	2.87	0.21
B1	Pressure Pattern	1.27	0.83	1.46	0.11	B3	Maximum Force	3.93	3.73	2.17	0.16
B1	Maximum Force	1.34	0.66	3.11	0.23	B3	Pressure Pattern	4.89	3.92	3.08	0.23
B2	Maximum Force	1.39	1.03	1.08	0.08	B4	Double Pulse	2.23	1.82	1.64	0.12
B2	Double Pulse	1.68	1.29	1.37	0.10	B4	Maximum Force	2.33	1.86	1.54	0.11
B2	Thumb Roll	1.74	1.37	1.19	0.09	B4	Thumb Roll	2.46	2.07	1.58	0.12
B2	Thumb Bob	2.08	1.34	2.18	0.16	B4	Thumb Bob	2.64	2.28	1.65	0.12

Table 5.4: Descriptive Statistics: *Mean*, *median*, *standard deviation* (SD) and *standard error* (SE) of ***task-completion time*** for *direction mechanisms* within *blocks* (B1,...,B4).

Direction mechanism had a *significant main effect* on *completion time* in context of *positional control* within *multiple regions*.

B3 Similarly, the *analysis* revealed a *significant main effect* of *direction mechanism* in context of *positional control* within *multiple regions*, $F(4{,}886) = 13.47, p < .00001$. In this manner, the *analysis* identified *thumb bob* ($M = 2.89, SD = 1.45$) to be *significantly faster* than *maximum force* ($M = 3.93, SD = 2.17$), $t(886) = 3.925, p < .001$. Interestingly, *pressure pattern* depicted the *slowest condition* ($M = 4.89, SD = 3.08$) and was *significantly slower* than *thumb bob* ($M = 2.89, SD = 1.45$), $t(886) = 6.998, p < .0001$, *double pulse* ($M = 3.10, SD = 1.71$), $t(886) = 5.121, p < .0001$ and *thumb roll* ($M = 3.70, SD = 2.87$), $t(886) = 3.806, p < .01$. However, *differences* between *pressure pattern* and *maximum force* were not *significant*, $t(886) = 3.074, n.s.$.

Direction mechanism had a *significant main effect* on *completion time* within B4.

B4 Finally, *direction mechanism* also showed a *significant main effect* in case of *rate-based control* when used with a *one-to-many mapping*, $F(3{,}707) = 4.47, p < .01$. *Post hoc pairwise comparisons* with a *Bonferroni correction* revealed that, *thumb bob* ($M = 2.64, SD = 1.65$) performed significantly *slower* than *double pulse* ($M = 2.23, SD = 1.64$), $t(707) = 3.443, p < 0.0037$. All *other differences* were not *significant*.

Number of Crossings

Number of crossings required a *non-parametric test*.

In contrast to the *previous response*, *number of crossings* represents *count data* and hence requires a *nonparametric test*, since a *traditional anova* does not apply. Consequently, we decided to utilize an *aligned rank transform* (ART), as suggested by Payton et al. [2006], yielding the *following results* according to *block* **B1**,...,**B4**:

Block	Direction Mechanism	Mean	Median	SD	SE	Block	Direction Mechanism	Mean	Median	SD	SE
B1	Thumb Bob	1.16	1.00	0.53	0.04	B3	Thumb Bob	1.17	1.00	0.48	0.04
B1	Thumb Roll	1.22	1.00	0.51	0.04	B3	Maximum Force	1.25	1.00	0.65	0.05
B1	Maximum Force	1.24	1.00	0.52	0.04	B3	Thumb Roll	1.33	1.00	0.72	0.05
B1	Double Pulse	1.27	1.00	0.63	0.05	B3	Double Pulse	1.49	1.00	0.84	0.06
B1	Pressure Pattern	1.47	1.00	0.81	0.06	B3	Pressure Pattern	1.56	1.00	0.76	0.06
B2	Maximum Force	1.06	1.00	0.23	0.02	B4	Double Pulse	1.14	1.00	0.43	0.03
B2	Double Pulse	1.06	1.00	0.26	0.02	B4	Thumb Roll	1.22	1.00	0.49	0.04
B2	Thumb Bob	1.08	1.00	0.31	0.02	B4	Maximum Force	1.26	1.00	0.67	0.05
B2	Thumb Roll	1.09	1.00	0.29	0.02	B4	Thumb Bob	1.28	1.00	0.58	0.04

Table 5.5: Descriptive Statistics: *Mean*, *median*, *standard deviation* (SD) and *standard error* (SE) of ***number of crossings*** for *direction mechanisms* within *blocks* (B1,...,B4).

B1 The *test* showed a *significant main effect* for *direction mechanism* on *crossings* in context of *positional control* within a single *multi-range region*, $F(4{,}886)=6.92, p<.00001$. This way, *pressure pattern* led to the *highest number of crossings* ($M=1.47, SD=0.81$) and was *significantly less accurate* than *thumb bob* ($M=1.16, SD=0.53$), $t(888.88)=5.144, p<.0001$, and *thumb roll* ($M=1.22, SD=0.51$), $t(888.88)=3.519, p<.01$, as revealed by *Post Hoc pairwise comparisons* with *Bonferroni correction*. However, all *other differences* were not *significant*.

Direction mechanism had a *significant main-effect* on *crossings* within B1.

B2 In contrast, there was no evidence to suggest that *direction mechanism* had an *effect* on *crossings* when using *rate-based control* in a *one-to-one mapping*, $F(3{,}707)=0.71, n.s.$. Hence, all *mechanisms* performed *equally accurate* (*maximum force* ($M=1.06, SD=0.23$), *double pulse* ($M=1.06, SD=0.26$), *thumb bob* ($M=1.08, SD=0.31$), and *thumb roll* ($M=1.09, SD=0.29$)).

Differences in terms of *crossings* were not statistically different within B2.

B3 However, further *analysis* revealed a *significant main effect* of *direction mechanism* in case of *positional control* with *multiple regions*, $F(4{,}886)=12.93, p<.00001$. *Post Hoc pairwise comparisons* with *Bonferroni correction* showed that *double pulse* ($M=1.49, SD=0.84$) led to *significantly more crossings* than *maximum force* ($M=1.25, SD=0.65$), $t(884.18)=3.560, p<.01$, and *thumb bob* ($M=1.17, SD=0.48$), $t(884.18)=4.369, p<.001$. Similarly, *pressure pattern* ($M=1.56, SD=0.76$) was the least *accurate* and had *significantly more crossings* than *maximum force* ($M=1.25, SD=0.65$), $t(884.18)=5.325, p<.0001$, *thumb bob* ($M=1.17, SD=0.48$), $t(884.18)=6.133, p<.0001$ and *thumb roll* ($M=1.33, SD=0.72$), $t(884.18)=4.078, p<.001$. Still, *other differences* were not *significant*.

Direction mechanism had a *significant main-effect* on *crossings* within B3.

Differences within B4 were not *statistically significant*.

B4 Finally, the *analysis* did not show an *effect* of *direction mechanism* when using *rate-based control* within *multiple regions*, $F(3,707) = 2.95, n.s.$. Still, *double pulse* ($M = 1.14, SD = 0.43$) was identified to have the *least number of crossings*.

Discussion

All direction mechanisms performed equally fast within B1.

Above-stated results are summarized in *tables 5.4* and *5.6*, including *mean*, *median*, *standard deviation* and *standard error* for both *responses* respectively. Interestingly, with respect to the *first block*, findings suggest that *choosing* one *direction mechanism* over the *other* yields similar *performance* in terms of *completion time*, and only *differentiates* in *number of crossings*. In this manner, *pressure pattern* caused *significantly more crossings* than *thumb bob* (1.39 vs. 2.08) and *thumb roll* (1.39 vs. 1.74), and hence is *not suited* to be used within *B1*. Nevertheless, note that all other *mechanisms* performed *similar*, and hence are *appropriate* to operate *interchangeably*. Clearly, this *result* was expected, since *acquiring* and *selecting targets* in context of *positional control* with a *one-to-one mapping* (**B1**), rarely required *participants* to *specify directions*. Consequently, *proposed direction mechanisms* only had to be used for *below-located targets* as well as *extreme points*.

All *direction mechanisms* obtained similar *levels of control*.

In contrast, *maximum force* was identified to *perform fastest* within the *second block* (**B2**) and showed the *least number of crossings*, together with *double pulse*. Nevertheless, it is important to realize that *differences* in *crossings* were not *statistically significant*, suggesting that all *mechanisms* achieved comparable *levels of control*. However, note that *thumb bob* performed *worst* in terms of *completion time* and was *significantly slower* than *maximum force*. *User feedback* revealed that *participants* perceived the *threshold* for *bobbing* as *too low*, resulting in *unintended changed*. Hence, *thumb bob* should be *avoided* when using *rate-based control* in a *single multi-range region*. Equally important, *double pulse* represents a *reasonable alternative* to *maximum force*, since *differences* were not *significant*. Nevertheless, even though *thumb roll* performed *slightly slower* than *maximum force*, it allowed *participants* to complete the *task* $0.34s$ *faster* than the *slowest condition*.

In addition, the *third block* required users to make use of *positional pumping* to acquire *targets* that are located outside the *current region*. With this in *mind* an interesting *result* emerged from the *data*. While *double pulse*, *thumb bob* and *thumb roll* performed *equally well* with *thumb bob* yielding the *best overall performance*, *maximum force* and *pressure pattern* performed *significantly slower*, with *pressure pattern* resulting in the *highest number of crossings*.

A possible explanation for *above-stated results* might be given by the fact that *both direction mechanisms* require *users* to quickly apply *maximum force* to *specify directions*. This observation is crucial, since **B3** makes use of *positional pumping*, where *pressure* is *repeatedly applied* until the *target segment* is reached. Nevertheless, note that *positional pumping* can not be performed *as quickly as possible*, since *maximum force* and *pressure pattern* are both sensible to *fast force increase*, and hence would *inevitable* result in *unintended direction changes*. Conversely, *thumb bob* allowed *participants* to quickly apply *maximum force* without *changing directions*, resulting in the *best overall performance*. Please be informed that these *findings* have further *strengthened* our decision to avoid *maximum force* and *pressure pattern*, when using *positional control* in context of *multiple regions*.

Interestingly: *maximum-force* and *pressure pattern* performed slower, since *direction mechanisms* collided with *positional pumping*.

Finally, *results* of the *last block* (**B4**) revealed that *direction mechanisms* did not differ among the *achievable level of control*, and also showed *similar performance* in terms of *completion time*. As a result, even though *double pulse* achieved the *best overall performance*, *maximum force* and *thumb roll* represent *well-suited alternatives*. However, please be aware that *thumb bob* performed *slowest* and also led to the *highest number of crossings*. Thus, *thumb bob* is not appropriate to be used within *B4*. Unfortunately, we have not looked into combining the *pressure pattern switch*, as introduced in *section 4.1.3*, with *rate-based control* as *pressure-control mechanism*. This way, although *participants* seemed to have *difficulties* when controlling *force* during *pressure release*, a *natural mapping* might yield *better performance* when *rate-based control* is applied. Hence, we are motivated to look into an *additional design* that enables *bidirectional force input* from a *static location*.

Findings suggest to also explore *rate-based control* when combined with *pressure pattern* as direction mechanism → additional bidirectional design.

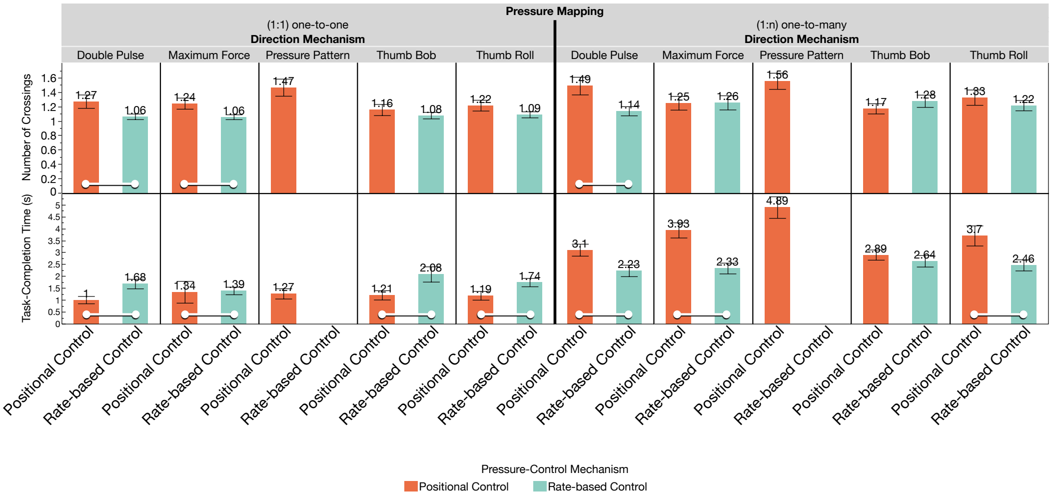


Figure 5.7: Bar-charts representing *means* in **T2**: bottom: *task-completion time*, top: *number of crossings* (*error bars:* 95% CIs, *connection lines:* significant difference).

We reject our *first hypothesis* (H1).

Having discussed *findings* of the *first test*, it is now possible to state that we *reject H1* and *accept* the *alternative hypothesis*. Consequently, *acquiring targets* with different *direction mechanisms* strongly affects *performance* for *fixed combinations* of *pressure mapping* and *pressure-control mechanism*. Subsequently, we draw the *reader's attention* to the *second test* to assess *H2*.

5.5.4 Second Test [T2]

In contrast to the *previous analysis*, the *second test* focusses on H2, and tries to identify which *pressure-control mechanism* performs *best* for given *combinations* of *pressure mapping* and *direction mechanism*. Note that the *responses' means* are illustrated in *figure 5.7*, along with *error-bars*, representing 95% *confidence intervals* (CIs). Moreover, *connection lines* between *bars* emphasize *conditions* that are *significantly different*. Indeed, *significance* was accepted at $\alpha = 0.05/8 = 0.00625$, since *eight comparisons* are made. Subsequently, *findings* for each of the *responses* along with a *discussion* are stated. Note that *results* are categorized by the *type* of *pressure mapping* being used, i.e., whether *one-to-one* [1:1], or *one-to-many* [1:n] is encountered:

Significance was accepted at $\alpha = 0.00625$.

Task-Completion Time

[1:1] Interestingly, the *second analysis* revealed that *pressure control mechanism* had a *significant main effect* on *completion time*, independent of the utilized *direction mechanism* (*figure 5.7*). Hence, *positional control* ($M = 1.00, SD = 1.05$) performed *significantly faster* than *rate-based control* ($M = 1.68, SD = 1.37$), when *double pulse* is applied, $F(1{,}349) = 57.41, p < .00001$. Moreover, in combination with *maximum force*, *positional control* ($M = 1.34, SD = 3.11$) performed *significantly faster* than *rate-based control* ($M = 1.39, SD = 1.08$), $F(1{,}349) = 35.09, p < .00001$. Nevertheless, please be aware that the *effect size* of $0.05s$ is *negligible small*. Similarly, also in case of *thumb bob controlling* the value's *absolute position* ($M = 1.21, SD = 1.33$) was identified to be *significantly faster* than adjusting the *speed* with which *values are changing* ($M = 2.08, SD = 2.18$), $F(1{,}349) = 52.14, p < .00001$. Remarkably, this *condition* also led to the *maximum effect size* of $0.87s$. Finally, in case that *thumb roll* is used, *positional control* ($M = 1.19, SD = 1.25$) allowed *participants* to complete the *task significantly faster* than *rate-based control* ($M = 1.74, SD = 1.19$), $F(1{,}349) = 68.11, p < .00001$.

Pressure-control mechanism had a *significant main-effect* on *completion time* in context of a *one-to-one mapping*,

[1:n] Surprisingly, when investigating *completion time* in context of *multiple regions*, the *converse result* emerged from the *data*. Hence, as shown in *figure 5.7*, *rate-based control* performed *consistently faster* than *positional control* independent of the utilized *direction mechanism*. This way, *velocity control* ($M = 2.23, SD = 1.64$) was *significantly faster* than *positional control* ($M = 3.10, SD = 1.71$) if *double pulse* is applied, $F(1{,}349) = 51.67, p < .00001$. In addition, in context of *maximum force*, *rate-based control* ($M = 2.33, SD = 1.54$) was revealed to be *significantly faster* than *positional control* ($M = 3.93, SD = 2.17$), $F(1{,}349) = 39.41, p < .00001$, yielding the *largest effect* of $1.6s$. Interestingly, *differences* in context of *thumb bob* were not *statistically significant* $F(1{,}349) = 0.39, n.s.$. Nevertheless, when *thumb roll* was used, *rate-based control* ($M = 2.46, SD = 1.58$) performed *significantly faster* than *positional control* ($M = 3.70, SD = 2.87$), $F(1{,}349) = 43.42, p < .00001$.

With respect to *multiple regions*, *rate-based control* performed significantly faster than *positional control*.

Number of Crossings

[1:1] Turning to *number of crossings* within a *single multi-range region*, the *analysis* only revealed significant *main effects* for *double pulse* and *maximum force*, as illustrated in *figure 5.7*. In this manner, *positional control* ($M = 1.27, SD = 0.63$) led to *significantly more crossings* than *rate-based control* ($M = 1.06, SD = 0.26$), $F(1,349) = 21.01, p < .00001$. Similarly, if *maximum force* is applied, *positional control* ($M = 1.24, SD = 0.52$) caused *significantly more crossings* than *rate-based control* ($M = 1.06, SD = 0.23$), $F(1,349) = 20.07, p < .00001$. Still, all other *differences* were not significant.

[1:n] In contrast, with respect to *multiple regions*, *pressure control mechanism* only had a *significant main effect* on *crossings* in case of *double pulse* (*figure 5.7*). In this regard, *rate-based control* ($M = 1.14, SD = 0.43$) caused *significantly less crossings* than *positional control* ($M = 1.49, SD = 0.84$), $F(1,349) = 30.12, p < .00001$, resulting in an *overall effect size* of 0.35. Nevertheless, it is important to realize that *none* of the *remaining differences* was *statistically significant*.

Discussion

Positional control performed *fastest* within a *one-to-one mapping*.

Having referred to *results* of the *second test*, there is *strong evidence* to suggest that *positional control* performs *better* in terms of *completion time*, when used in context of *[1:1]-mappings*. This way, *targets* could be *selected* more *quickly*, as soon as *positional control* was applied. However, note that in case of *maximum force*, *differences* can be ignored, since the *effect size* is *negligible small*.

Rate-based control performed *better* in terms of *crossings* within a *one-to-one mapping*.

In contrast, when referring to *number of crossings*, we see an *interesting effect*. *Rate-based control* led to *strictly less crossings* independent of the utilized *direction mechanism*. Hence, even though *differences* were only *significant* in case of *double pulse* and *maximum force* (*figure 5.7*), it is *reasonable* to suggest that *rate-based control* offers *better control* within *[1:1]-mappings*. Hence, we identify a *trade-off* between *positional-* and *rate-based control*, i.e., *speed* vs. *accuracy*.

Turning to *[1:n]-mappings*, *findings* suggest that *rate-based control* is the *best choice* in terms of *performance*, since it was *faster* among all *direction mechanisms*, and led to *significantly less crossings* for *double pulse*, while showing *comparable results* for *remaining techniques*. Equally important, *differences* among *pressure-control mechanisms* in terms of *completion time* were all *statistically significant*, except for *thumb bob*. Indeed, a *possible explanation* might be that *rate-based control* allowed *participants* to make *quick-progress* among *multiple regions* and does not rely on an *auxiliary mechanism*, like *positional pumping*. In this manner, *long-distance targets* are *easily accomplished* without having to deal with *restrictions* as caused by the *finiteness* of the *force-sensitive range*. Consequently, we reject *H2*, as stated in *section 5.1*.

Rate-based control performed *best* within *one-to-many mappings*, since it does not rely on *positional pumping*, and hence allows for *quick progress* among *multiple regions*.

H2 is rejected.

5.5.5 Questionnaire

Finally, we draw the *reader's attention* to the *third hypothesis* (*section 5.1, H3*) to assess whether *direction mechanisms* are equally liked by *participants*. Hence, the *following sections* include an *analysis* for each *block* respectively, together with *comments* and *suggestions*, we have obtained in the *study*.

Analysis

B1 A *Friedman-test* was run to determine whether *participants* preferred *one direction mechanism significantly more* than *another* when *positional control* within a *one-to-one mapping* was used. Note that *user-preference* was measured on a *seven-point likert-scale*, as illustrated in *figure B.1*, ranging from *totally disagree* [1] to *totally agree* [7]. Interestingly, *user-preference* was *statistically different*, $\chi^2(4) = 15.160, p < .01$. *Post hoc pairwise comparisons* (SPSS Statistics, 2017) with a *Bonferroni correction* revealed that *thumb bob* ($M = 3.50, Mdn = 3.50, SD = 1.90$), was *significantly less preferred* than *maximum force* ($M = 5.80, Mdn = 6.0, SD = 0.79$) ($p < .05$) and *thumb roll* ($M = 6.10, Mdn = 6.50, SD = 0.99$) ($p < .05$), and hence was the *least preferred*. Nevertheless, note that *remaining differences* were not *significant*.

User-preference was statistically different among *direction mechanisms* in B1.

DM	PCM	PM	T	B	Mean	Mdn	SD	Min	Max	PM	T	B	Mean	Mdn	SD	Min	Max
Maximum Force	PC	1:1	T1	B1	5.80	6	0.79	5	7	1:n	T10	B3	5.10	5	1.45	3	7
Thumb Bob	PC	1:1	T2	B1	3.50	3.5	1.90	1	7	1:n	T11	B3	4.60	4.5	2.01	1	7
Thumb Roll	PC	1:1	T3	B1	6.10	6.5	0.99	5	7	1:n	T12	B3	4.90	5	1.37	2	6
Double Pulse	PC	1:1	T4	B1	4.60	5	1.58	2	6	1:n	T13	B3	5.20	5.5	1.03	3	6
Pressure Pattern	PC	1:1	T5	B1	4.10	4	1.91	2	7	1:n	T14	B3	3.30	3	1.64	1	5
Maximum Force	RbC	1:1	T6	B2	6.30	6.5	0.82	5	7	1:n	T15	B4	5.60	6	0.97	4	7
Thumb Bob	RbC	1:1	T7	B2	4.00	4	1.70	1	7	1:n	T16	B4	4.20	4	1.87	1	7
Thumb Roll	RbC	1:1	T8	B2	5.70	6	0.95	4	7	1:n	T17	B4	5.10	5	1.20	3	7
Double Pulse	RbC	1:1	T9	B2	5.30	6	1.25	3	7	1:n	T18	B4	4.90	5.5	1.45	2	6

Table 5.6: Questionnaire Statistics: *direction mechanism* (DM), *pressure-control mechanism* (PCM), *pressure mapping* (PM), *technique* (T), *block* (B), *mean*, *median*, *standard deviation* (SD), *min* and *max* of ***user-preference*** measured on a *seven-point likert-scale*.

User-preference was statistically different among *direction mechanisms* in B2.

B2 Similarly, an additional *Friedman-test* revealed that *differences* within B2 are *statistically significant*, $\chi^2(3) = 12.419, p < .01$. *Pairwise comparisons* were performed (SPSS Statistics, 2017) with *Bonferroni correction* for *multiple comparisons*. In this manner, a *post hoc analysis* revealed that *thumb bob* ($M = 4.0, Mdn = 4.0, SD = 1.70$) was *significantly less preferred* than *maximum-force* ($M = 6.30, Mdn = 6.50, SD = 0.82$), but still received *average results*. All *other differences* were not *significant*.

User-preference was not statistically different within B3 and B4.

B3 In contrast to *previous blocks*, *user-preference* within B3 was not *significantly different*, $\chi^2(4) = 8.181, p = .085$.

B4 Finally, also in case of *rate-based control* within *multiple regions*, there was no evidence to *suggest* that *direction mechanism* had an effect on *user-preference*, $\chi^2(3) = 5.370, p = 0.147$.

H3 is rejected.

Above-stated preference-results motivated us to *reject* H3. However, please be informed that we also assessed *user's overall preference* by asking *participants* to rank each *mechanism* from 1 to 5, where 1 referred to the *highest*, and 5 to the *lowest ranking*. Indeed, *participants* were allowed to *assign* the *same ranking* to *multiple techniques*. Interestingly, *preference* was *significantly different*, $\chi^2(4) = 9.548, p < .05$. *Post hoc pairwise comparisons* with *Bonferroni correction* revealed that *thumb roll* ($M = 1.90, Mdn = 1.50, SD = 1.10$) was *significantly more preferred* than *pressure pattern* ($M = 3.90, Mdn = 4.50, SD = 1.45$), ($p < .01$). Nevertheless, all *other differences* were not *significant*. Subsequently, *comments* and *suggestions* are stated, to provide *further insights* into *users' personal preference* in addition to *above-stated results*.

Comments and Suggestions

Throughout the *study* we obtained the *following feedback:*

- *Participants* noted that the *resting threshold* for *thumb bob* was *too low*, and hence led to *accidental activation*.
- In addition, *users* remarked that *pumping* among *multiple regions* with *pressure pattern* or *maximum force* required them to be *cautious*, since *applying maximum force* overlapped with *mechanisms' trigger*.
- Moreover, *two users* mentioned that *pressure pattern* caused *initial confusion*, since it was the *only mechanism* that utilized *pressure release* to *navigate* in the *opposite direction*, and hence was perceived as being *more difficult* than *applying pressure* from *zero-force*. Still, *users* developed *strategies* to *successfully complete* the *task* by *quickly applying maximum force*, followed by *immediate pressure release*. This way, the *cursor* dropped *multiple times*, until the *bottom* of the *target-region* was reached. Finally, *force application* led to the *intended location*.

The reversed mapping of *pressure pattern* led to confusion.

- Equally important, *participants* commented that having to apply *maximum force*, each time the *direction* needs to be changed, was *exhausting* over time.

Applying *maximum-force* was tedious over time.

- Similarly, even though *double pulse* was overall liked by *participants*, it was sometimes not recognized, since *users* occasionally *drifted* from the *predefined rhythm*.
- Finally, *users* appreciated *thumb roll*, since it provided *immediate access* to *both directions*.

Having discussed *quantitative*- as well as *qualitative results* of the *study*, we draw a *conclusion* and point to *resulting implications* for the *second study* that is part of *Chapter 6*.

5.6 Conclusion and Implications

In this *study* we have continued with our *systematic procedure*, as introduced in *Chapter 2* to come up with a *solution* to the *bidirectional problem*. In this regard, we have conducted *two tests*, as well as an *evaluation* of *qualitative data* to identify which combination of the *three essential components* (*section 4.1*) *performs best* and is *most preferred* by *participants*.

Findings suggest to only consider *one-to-many* and exclude *one-to-one* from further investigations.

Findings have led us to the *conclusion* to only consider *designs* that are built from *[1:n]-mappings* (*figure 5.4, B3 and B4*), and exclude *[1:1]* from *further considerations*. Note that this *decision* is justified by *results* of the *first test* (*section 5.5.3*) that revealed that *direction mechanisms* do not have an *effect* on *user-performance* in terms of *completion time*, since *users* are still *fast* when always *navigating* from *zero-force*. Consequently, we decided to only focus on the *more general case*, i.e., *[1:n]-mappings* where *bidirectional force input* is required, to *navigate* within *large sets* of *values*.

Positional control is omitted from *further considerations*.

In addition, note that we decided to focus on *rate-based-*, rather than *positional control* within *[1:n]-mappings*, since *results* of the *second test* (*section* 5.5.4) have revealed that *positional pumping* was *tedious* and led to *significantly more crossings* than *adjusting* the *speed*, with which *values* are changing. Interestingly, these *findings* are also in line with *previous results* of Ng and Brewster [2016] as well as Wilson et al. [2011], who identified *rate-based control* to outperform *positional control* in *mobile scenarios*, like *walking* or *driving*. In this regard, *results* suggest that *rate-based control* is *well-suited* for the *application scenario*, as introduced in *section 2.1*, where *users* have to *operate* their smartphone *single-handed* and only using their *thumb*. Hence, we are left with *bidirectional designs* contained within B4.

Considering *performance-* as well as *preference-results* within B4, *differences* among *double pulse*, *maximum force* and *thumb roll* were not *statistically significant*, suggesting that *all three designs* can be used *interchangeably*. In contrast, *thumb bob* was *significantly slower* than *double pulse* and hence is omitted from *further considerations*. Moreover, even though *double pulse* performed best, *users* occasionally had *issues* to perform the *predefined rhythm* precisely, resulting in *usability issues*. Similarly, even though *maximum force* was *liked* by *participants*, having to apply *maximum force*, each time the *direction* needs to be *changed*, was *exhausting* over time. Hence, we decided to *merge* the *advantages* of *both designs* into *quick pulse*, i.e., an *additional bidirectional design* that depicts a *simplified version* of the *double-pulse gesture*, and only requires to *reach* the *center* instead of the *maximum* of the *force-sensitive range*. Moreover, we also decided to explore *thumb roll* without *further adjustments*.

Maximum-force and *double-pulse* merge into *quick pulse*.

Design	Description
Quick Pulse	Quickly perform a *dominant pulse* to *toggle directions*.
Thumb Roll	Roll either *left* or *right* to move in the *respective direction*.
Natural Mapping	*Increase:* just exert *pressure*, *decrease:* quickly apply maximum force, followed by *slow pressure-release*

Table 5.7: *Remaining designs* of the *initial set* of *eighteen designs* that are *further explored* in the *second study* (*Chapter 6*).

Finally, we also got inspired to *look* into another *combination*, namely *pressure pattern* combined with *rate-based control*, that has not been explored in this *study*. In this regard, even though *pressure pattern* showed *bad performance* when used with *positional control*, we recognized that the *comparison* with *remaining designs* was *unfair*, since it was the *only design* that utilized *pressure release* to *navigate* in the *opposite direction*. In addition, note that *pressure pattern* could be *easily confused* with *maximum force*, since both required to reach the *maximum* of the *force-sensitive range* to *specify directions*. Hence, we are *motivated* to look into an *additional design*, i.e., *natural mapping*, where *users* do not adjust the *value's absolute position*, but rather the *value's changing speed* in accordance to the *amount of force* that is *released*.

Findings suggest an *additional design*, i.e., *natural mapping*.

Consequently, we obtain *three* out of the *initial set* of *eighteen designs*, as illustrated in *table 5.7* that are *further explored* in the *second study* by comparing their *performance* against a *baseline condition*.

Chapter 6

Second Study

In the *beginning of this* thesis, *force input* has been identified to be *well-suited* to overcome *reachability*- and *occlusion issues* of the *human thumb*, during *single-handed smartphone operation*. However, please be reminded that *force* is limited in the *way* that it is *unidirectional* (*Chapter 2*). To tackle this *issue*, we followed a *systematic procedure* and identified *eighteen interaction designs* to enable *bidirectional force input* from a *static location*. While the *first study* examined which *combination* of *pressure mapping*, *direction*- as well as *pressure-control mechanism* performs *best* and is *most preferred*, the *second study* concentrates on *remaining designs*, namely *quick pulse*, *thumb roll* and *natural mapping*, by evaluating their *performance* against a *baseline condition*. *Findings* allow us to *shed light* on the *second research question*, as introduced in *section 2.6*, and identify the *design* that performed best in terms of *user-preference* and *performance*.

The second study focusses on *remaining designs*, and evaluates their performance against a *baseline condition*.

Please be informed that the *study's task* mostly corresponds to the *target-acquisition* and *selection task*, as used in the *previous study* (*section 5.2*). Hence, the *following sections* only deal with *important changes* regarding *hypotheses*, the *study's task*, as well as *design decisions* made. In addition, *changes* according to the *experimental design* are discussed. Finally, the *chapter* concludes with *study results*, and draws a *conclusion* regarding our *research questions*.

The *study's task* is similar to the *target-acquisition* and *selection-task* of the *previous study*.

6.1 Hypotheses

Throughout the *study*, we examine the following *hypotheses* (stated in *null form*, i.e., expected to be *rejected*):

H1 For any *technique*, *targets* are selected at the *same speed*, i.e., *task-completion time* is independent of *technique*.

H2 For any *technique*, *targets* are passed equally often, i.e., *number of crossings* is independent of *technique*.

H3 For any *technique*, proper *targets* are chosen, i.e., *target-accuracy* is independent of *technique*.

H4 *Techniques* are *equally liked* by *participants*, i.e., *user-preference* is independent of *technique*.

6.2 Techniques

The *previous study* identified *three bidirectional designs*, namely *quick pulse*, *thumb roll* and *natural mapping*, whose *evaluation* is the *main objective* of this study. Hence, we briefly remind the *reader* of the *functional concepts* of each of the *above-stated designs* by referring to the *intended use-case*, as introduced in *section 4.3.1*. Note that *subsequent explanations* only consider *quick pulse* and *natural mapping* rather than *thumb roll*, since *thumb roll* was already described in *section 4.3.2*. Equally important, we do not *explicitly state* the type of *pressure-mapping* and *control-mechanism* anymore, but only refer to the *direction mechanism's name* to reference *designs*. Consequently, *thumb roll* serves as an *abbreviation* for *one-to-many thumb roll rate-based control*.

Subsequently, techniques are only identified by the *direction mechanism's name*.

Quick Pulse The *first design* was obtained by merging *double pulse* and *maximum force*. In this regard, *correcting* an *undershoot* in *scenario A* (*figure 4.8*), i.e., from A^- to A', only requires *users* to *press slightly stronger*. In contrast, *overshoots*, i.e., from A^+ to A', are corrected using a *simplified version* of the *double pulse gesture* in form of a single *dominant pulse*. Conversely, *undershoot-corrections* in *scenario B*, i.e., from B^- to B', are made by exerting *slightly more force*, while *overshoot-corrections*, i.e., from B^+ to B', require *users* to *toggle directions*, followed by slow *pressure-increase*.

The direction is toggled by a *quick dominant pulse*.

Figure 6.1: *Target-acquisition* and *selection task* of *Study 2*.

Natural Mapping As opposed to the *previous technique*, *natural mapping* utilizes a *switch-mechanism*, and hence offers *immediate access* to *both directions*. This way, *undershoot-corrections* in *scenario A* (*figure 4.8*), i.e., from A^- to A', are made by further *exerting force*. In contrast, *users* can *move* in the *opposite direction*, i.e., from A^+ to A', by quickly applying *maximum force*, followed by slow *pressure-release*. In this manner, the *speed* with which *values* are changing is *specified* by the *amount of force* that is *reduced*. Likewise, *undershoots* within *scenario B* (*figure 4.8*), i.e., from B^- to B', are adjusted by slow *pressure-release*. In contrast, *overshoot-corrections*, i.e., from B^+ to B', require *users* to return to their *resting-threshold* and *re-exert force* until B' is met.

Pressure release is mapped to the *speed* with which the *value* decreases.

6.3 Task

Similar to the *previous study*, we adapted a *target-acquisition* and *selection task*, as illustrated in *figure 6.1*, to *assess* differences in *performance* of *above-stated techniques*. Nevertheless, note that *comparisons* are not only drawn among *bidirectional designs*, but also with respect to a *baseline condition* with which people are *already* familiar. Hence, the *task* was designed to utilize a *picker-representation* that could either be controlled using *force input* from a *static location* or *multi-touch*, to select *discrete-values* out of a *predefined range*. In this manner, *users* were asked to navigate to the *intended location*

A picker was used that could be either controlled with *force-* or *multi-touch input*.

Visual feedback about the *current direction* is provided by an *arrow* next to the *picker*.

as *quickly* and *accurately* as possible by exploiting the *technique announced* on-screen. Indeed, *visual feedback* about the *current direction* is provided at any time by the *direction indicator*, placed next to the *picker* (*figure 6.1*). Finally, *selections* are made, as soon as the *user's thumb* is lifted from *screen*.

The following is kept from Study 1:
test-trials
procedure
setting
design explorer
trial-widget

Please be aware that apart from the *new visualization* in form of the *value picker*, the *different selection mechanism* and *refined transfer-function*, other *components*, like *test-trials*, the *study's procedure*, *setting*, *design explorer* and *trial-widget* are maintained from the *previous study* (*section 5.2*). Hence, the *following section* only deals with *additional decisions* of the *second study's design*.

6.3.1 Task Design Decisions

The second study used a *picker* rather than a *slider* representation.

Value Picker In contrast to the *previous study*, we decided to choose a *picker-* rather than a *slider-visualization*. Note that this *decision* is justified by the fact that *sliders* are predisposed to *slight variations* while *lifting* the *thumb*, and hence are *inappropriate* to select *individual items* out of *large value domains* [Harley, 2015]. In addition, the *picker's footprint* is *significant smaller* than the *one* of a *slider*, since the *picker's cursor* remains *static* at a *centered location*, while the *content* is moved underneath it. As a result, only a *small portion* of the *value range* is exposed to the *user*, resulting in *lower screen-space requirements*.

The snapping-mechanism ensures that only *discrete-values* are selected.

Figure 6.2 illustrates the *picker*, as used in this *study*, measuring $394pt \times 216pt$ in size. While the *user's selection* is defined by the *value* that shares the *largest area* with the *cursor*, the *target* is highlighted in *green* (*figure 6.2*). Consequently, *participants* have to *navigate* to the *desired location* until *both values* correspond to each other. Equally important, *feedback* about the *picker's continuous location* is provided at any time, even though only *discrete-selections* are allowed. In addition, a *snapping-mechanism* is included to automatically adjust the *picker's content offset* to match the *position* of the *closest value*. As a result, the picker never stops in between, but only selects *discrete values* out of the *predefined range*.

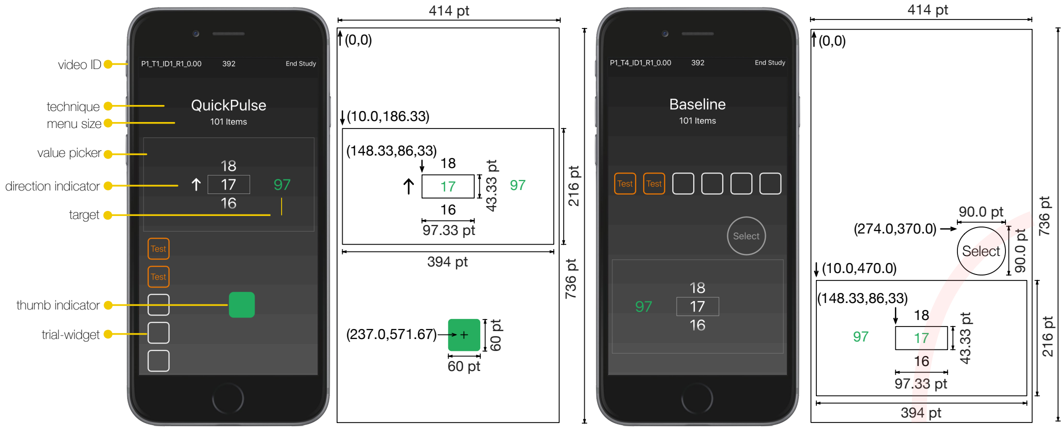


Figure 6.2: Left: user-interface for *quick pulse*, *thumb roll* and *natural mapping*, Right: user-interface for *baseline-condition* (*incl. interface dimension*).

Baseline In case of the *baseline-condition*, we decided to reposition the *picker* to be *placed* within the *thumb's reach* (*figure 6.2, right*). In this manner, *users* can navigate through the *value-range* by *sliding* directly on top of the *picker*. Nevertheless, please be aware that *sliding* is also possible when starting within the *picker's area* and *moving* outside its *boundaries*. In this regard, we adapt the *behavior* of the *standard iOS*™ *picker* to assure *fair comparisons*. Equally important, a *button* is included to finalize *selections*. In this regard, considerable care had to be taken to ensure that the *button* is placed within the *thumb's interaction range* (*figure 6.2, right: red line*), and is only enabled if the *picker* stopped *moving*.

For the *baseline condition*, the *picker* was placed within the *thumb's reach*.

Selection Mechanism Moreover, we decided to choose *quick-release* rather than *dwell-time* as *selection mechanism* (*section 2.4.3*) to eliminate *artificial delays* and *speed-up* the *study's overall procedure*. Consequently, *values* are chosen, as soon as the *user's thumb* is lifted from *screen*. Indeed, the *implementation* of *quick release* in context of *rate-based control* did not cause any *issues*, since *values remain constant*, as soon as *force* is no longer applied.

Quick-release rather than *dwell-time* was used.

Transfer Function Finally, we decided to *modify* the *transfer function* of the *previous study* to account for up to 101 *values* in the *longest condition*. In this manner, we applied the *original function* to 75% of the *force-sensitive range*, and assigned *faster speeds* to the *remaining part* of the *area*.

Note that this *design decision* is motivated by *research* conducted by Antoine et al. [2017], who assigned *faster speeds* to the end of the *force-sensitive range* to enable *quick progress* while *scrolling* towards *long-distance targets*. As a result, we obtain $T:[0,6.67]\rightarrow[0.0,14.79317665]$, mapping *force* (*section 4.4.1*) to *speed* [*mm/s*] with:

$$T(x) := \begin{cases} 0.0\,\textit{mm/s} & x \in [0, 1.35) \\ 12.40601507x - 16.748120337\,\textit{mm/s} & x \in [1.35, 5.34) \\ 19.411981086x + 23.29382565\,\textit{mm/s} & x \in [5.34, 6.67] \end{cases}$$

6.4 Design

6.4.1 Independent Variables (Factors)

Throughout the *study* we control the *following* factors:

Technique *Technique* depicts the *main factor* of this *study* and is controlled to assess *differences* in performance of *bidirectional designs*. In this regard, we distinguish *four* different *techniques*, namely *quick pulse*, *thumb roll* and *natural mapping*, as well as the *baseline-condition* to classify *performance* of *above-stated designs*.

Menu size corresponds to the *number of items* among which *values are chosen*.

MenuSize In addition, we control *menu size* to specify the *number of items*, among which *predefined targets* are chosen. While $[0, 9]$ depicts the *range* of a *calculator*, $[1, 30]$ and $[1, 60]$ refer to *days* and *seconds* respectively. Finally, $[0, 100]$ is used as the *standard range* to represent *percentage information*.

Direction Moreover, *direction* specifies whether *targets* are located *above* or *below* the *initial location*. As a result, *direction* affects the *frequency* with with *direction mechanisms* are used.

Relative distances are interpreted to the *menu size* that is currently set.

TargetDistance Finally, *target distance* determines how far *start-* and *target-position* are apart from each other. In this manner, *levels* include an *absolute step* of *one*, as well as three *relative distances*, namely *small20*, *medium50* and *large80*. Note that *relative distances* are interpreted according to the *menu size* that is currently set. As an example, given a *size* of $[1,60]$, *large80* corresponds to a *target distance* of $60\times0.8=48$.

Levels of *above-stated factors* are summarized in *table 6.1*.

factor	levels
Technique	*quick pulse, thumb roll, natural mapping, baseline*
MenuSize	[0,9], [1,30], [1,60], [0,100]
Direction	*above, below*
TargetDistance	*absoluteOne, small20, medium50, large80*

Table 6.1: Levels of *independent variables* in *Study 2*.

Even though *technique* is *determined* by announcing its *name* on-screen, *remaining factors*, i.e., *menu size*, *direction* as well as *target distance*, are *indirectly controlled* by specifying *start* and *target* respectively. *Table 6.2* provides an overview of all *values*, used in this *study*. In this regard, $x \rightarrow y$ corresponds to the *task* to select *y* as *quickly* and *accurately* as possible when starting from *x*. Clearly, an *infinite target width* is omitted, since it would otherwise confound our *results*.

Menu size, *direction* and *target distance* are controlled by specifying *start*- and *target-values* respectively.

		[0,9]	[1,30]	[1,60]	[0,100]
absoluteOne	above	$6 \rightarrow 7$	$19 \rightarrow 20$	$39 \rightarrow 40$	$67 \rightarrow 68$
	below	$4 \rightarrow 3$	$13 \rightarrow 12$	$26 \rightarrow 25$	$44 \rightarrow 43$
small20	above	$1 \rightarrow 3$	$3 \rightarrow 9$	$7 \rightarrow 19$	$11 \rightarrow 31$
	below	$6 \rightarrow 4$	$19 \rightarrow 13$	$39 \rightarrow 27$	$67 \rightarrow 47$
medium50	above	$3 \rightarrow 8$	$10 \rightarrow 25$	$20 \rightarrow 50$	$34 \rightarrow 84$
	below	$7 \rightarrow 2$	$23 \rightarrow 8$	$46 \rightarrow 16$	$78 \rightarrow 28$
large80	above	$0 \rightarrow 8$	$1 \rightarrow 25$	$1 \rightarrow 49$	$0 \rightarrow 80$
	below	$9 \rightarrow 1$	$30 \rightarrow 6$	$60 \rightarrow 12$	$100 \rightarrow 20$

Table 6.2: *Start* and *target* for *combinations* of *menu size*, *target distance* and *direction* ($x \rightarrow y \mathrel{\widehat{=}}$ select *y*, starting from *x*).

6.4.2 Dependent Variables (Measures)

In contrast, *dependent variables* remained unchanged. In this manner, *task-completion time* [s], *number of crossings* [count], *target accuracy* [true, false] and *user-preference* [7-point likert-scale] are logged throughout the *study*. Consequently, the *data format* of the *previous study* (*section 5.4*) only required the *following adjustments*:

Columns containing *independent variables* (**IVs**):
technique $\in$ {quick pulse, thumb roll, natural mapping, baseline}
menuSize $\in \{[0,9], [1,30], [1,60], [0,100]\}$
direction $\in$ {above, below}
targetDistance $\in$ {absoluteOne, small20, medium50, large80}

Table 6.3: Data format *adjustments* for *Study 2*.

6.4.3 Experimental Design

The *study* contains a $4 \times 4 \times 2 \times 4$ *factorial design*, as derived from the *following conditions*:

- 4 techniques (*quick pulse, thumb roll, natural mapping, baseline*)
- 4 menu sizes ($[0, 9], [1, 30], [1, 60], [0, 100]$)
- 2 directions (*above, below*)
- 4 target distances (*absoluteOne, small20, medium50, large80*)
- 3 repetitions for each condition
- 2 test trials per technique

Consequently, each *participant* performs $(4 \times 4 \times 2 \times 4 \times 3) + (4 \times 2) = (384 + 8) = 392$ *trials*, resulting in a *total duration* of $(392 \times 10s)/60s = 65.33min$, assuming an *average duration* of $10s$ per *trial*. Equally important, levels of *technique* and *menu size* are operated *sequentially*, to avoid *potential biases* caused by *frequent switching* among *multiple conditions*. Nevertheless, please be aware that the *order* in which *levels* are encountered is counterbalanced with 4×4 latin squares. As a result, care had to be taken that the *number* of *participants* depicts a *multiple* of *four*. In contrast, *levels* of *remaining factors* were *fully randomized*. Having referred to the *study design*, including *task*, *factors* and *measures*, the *chapter* concludes with *important results* obtained in this *study*.

Technique and menu size are kept constant to avoid confusion. All other conditions are *fully randomized*.

6.5 Study Results

6.5.1 Procedure

Sixteen *right-handed participants*, *five* of them *female* and aged between 21 and 31 ($M = 26.19, SD = 2.71$), participated in the *study*, yielding an *overall dataset* of $392 \times 16 = 6272$ *trials*. *Test trials* as well as 32 *outliers* were removed, resulting in $6272 - (4 * 2 * 16) - 32 = 6112$ *measurement points*. Since the *study* is meant to shed light on the *second research question* (*section 2.6*) to identify the *technique* that *performs best* and is *most preferred*, we only examined *main effects* of *technique* on each *measure* respectively. Consequently, *main effects* of *menu size*, *target distance* and *direction* are omitted, since they would consider *multiple techniques* at once.

We only examined *main-effects* for *technique*.

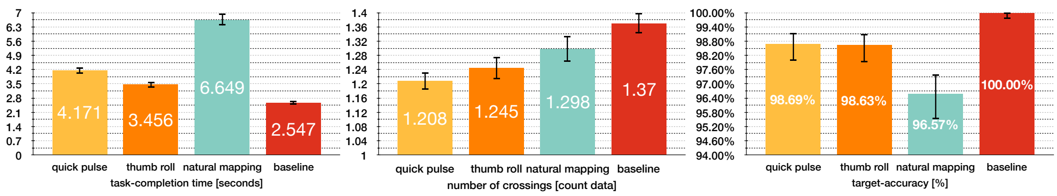


Figure 6.3: Descriptive Statistics of *Study 2*: means of *task-completion time* [*seconds*], *number of crossings* [count data] and *target-accuracy* [%] (*error bars:* 95% CIs).

Unfortunately, *task-completion time* was not *normally distributed*, as assessed by *visual inspection* of *normal Q-Q plots*. Nevertheless, a *log-transformation* allowed us to *run* a *repeated measures anova* on the *transformed data*. In contrast, *number of crossings* required *Friedman's* and *Wilcoxon's signed rank tests* respectively, since *count data* is not *suited* for an *analysis of variance*. Finally, *nonparametric tests*, i.e., *Cochran's Q* and *Mc Nemar tests*, were applied, since *target-accuracy* forms a *bi-partition* such that *users* could have either *acquired* or *missed* the *intended target*. In this manner, *performance* was measured in terms of *completion time*, *number of crossings* and *target-accuracy*, while *participant's personal preference* was assessed through *rankings* at the *end* of the study. Subsequently, *findings* for each of the *measures* are stated.

Completion time was *log-transformed* to run an *analysis of varience*.

6.5.2 Analysis

Task-Completion Time

The *analysis* revealed a *significant main-effect* of *technique* on *task-completion time*, $F(2{,}6065) = 955.7348, p < .0001$. *Post hoc pairwise comparisons* with *Bonferroni correction* showed that *differences* between all *techniques* are *statistically significant* ($p < .0001$). As illustrated in *figure 6.3*, *natural-mapping* ($M = 6.649, SD = 5.572$) performed *slowest*, followed by *quick pulse* ($M = 4.171, SD = 3.047$), *thumb roll* ($M = 3.456, SD = 2.509$) and *baseline* ($M = 2.548, SD = 1.820$). Hence, the *latter* depicts the *fastest condition*, and was *910ms* quicker than *thumb roll*. Clearly, these *results* are expected, since *participants* are already *familiar* with *multitouch-input*. Hence, returning back to our *hypotheses* of *section 6.1*, we reject H_1.

Technique had a *significant main-effect* on *task-completion time*.

H1 is rejected.

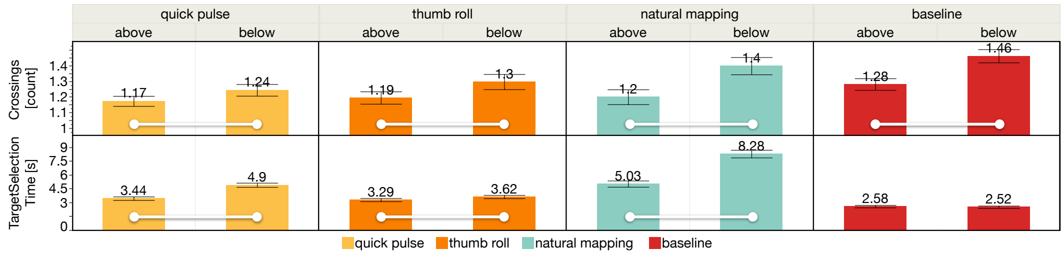


Figure 6.4: Descriptive statistics: means of *task-completion time* [*seconds*] and *number of crossings* [count data] according to **technique** and **direction** (*error bars:* 95% CIs).

			menu size				**target distance**				**direction**	
			[0,9]	[1,30]	[1,60]	[0,100]	absOne	small20	med50	large80	above	below
completion time [s]	Quick Pulse	M	2.6845	3.7643	4.6906	5.5612	1.8631	3.8988	4.9669	5.9636	3.4429	4.9020
		SD	2.1479	2.5300	2.9505	3.5987	1.6274	2.7136	2.8697	3.1154	2.7106	3.1892
	Thumb Roll	M	1.9827	2.9633	4.0208	4.8627	1.3203	2.9879	4.3426	5.1874	3.2888	3.6219
		SD	1.1845	1.8149	2.4961	3.1009	1.0179	1.6940	2.5140	2.5390	2.4701	2.5382
	Natural Mapping	M	4.5422	5.6990	7.6397	8.7047	3.9666	5.7029	7.8092	9.1626	5.0259	8.2837
		SD	4.6355	4.3906	5.8631	6.2129	4.6653	4.8123	5.4141	5.8758	4.7719	5.8389
	Baseline	M	1.4345	2.1476	3.0036	3.6057	0.8552	2.1508	3.1801	4.0087	2.5786	2.5165
		SD	0.6743	1.2044	1.8568	2.2794	0.6040	0.9607	1.5607	1.9958	1.8856	1.7528
crossings [count]	Quick Pulse	M	1.1563	1.1906	1.2193	1.2684	1.0574	1.2827	1.2370	1.2572	1.1736	1.2435
		SD	0.4414	0.4607	0.5054	0.5353	0.2743	0.5963	0.4723	0.5201	0.4514	0.5205
	Thumb Roll	M	1.1932	1.2083	1.2656	1.3150	1.0703	1.2526	1.3238	1.3360	1.1948	1.2960
		SD	0.5916	0.5721	0.6151	0.6962	0.3653	0.5747	0.7160	0.7273	0.5534	0.6796
	Natural Mapping	M	1.3368	1.2667	1.2891	1.2989	1.2958	1.2711	1.3166	1.3085	1.1997	1.3968
		SD	0.8233	0.7039	0.6758	0.6778	0.7238	0.6272	0.7834	0.7487	0.6626	0.7660
	Baseline	M	1.3525	1.3438	1.3629	1.4204	1.0208	1.5405	1.4648	1.4543	1.2803	1.4595
		SD	0.5497	0.5227	0.5710	0.5952	0.1430	0.5859	0.5951	0.6117	0.5170	0.5877

Table 6.4: Descriptive Statistics: *mean* (M), *standard deviation* (SD) for *task-completion time* and *number of crossings* according to *menu size*, *target distance* and *direction*.

Force techniques performed significantly faster for *above-located* targets.

Interestingly, the *analysis* showed a significant *technique* × *direction* interaction effect, $F(3{,}6065) = 163.6972, p < .0001$. *Post hoc pairwise comparisons* with *Bonferroni correction* revealed that *all techniques*, except *baseline*, performed *significantly faster* for *above-* than *below-located targets* ($p < .0001$, resp., *figure 6.4*). Indeed, this result is expected, since the *initial direction* was always set to *increase*. Equally important, *completion time* between *directions* varied the *least* for *thumb roll* (*figure 6.4*, $0.33s$) among all *force techniques*, since *users* had *immediate access* to *both directions*. However, even though *natural mapping* also represents a *switch-mechanism*, *differences* among *directions* were considerably *higher*, i.e., $3.25s$.

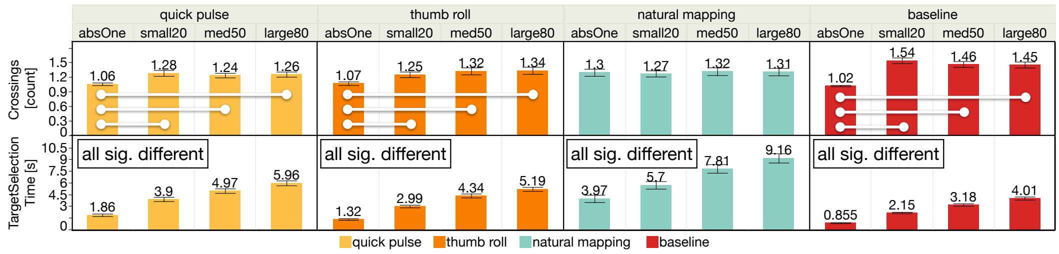


Figure 6.5: Descriptive statistics: means of *task-completion time* [*seconds*] and *number of crossings* [count data] according to **technique** and **distance** (*error bars:* 95% CIs).

Techniques performed *significantly different* for *below-located targets*.

In addition, *differences* between *techniques* for *below-located targets* were significant ($p < .05$, resp.). In this regard, *natural mapping* performed *slowest* ($M = 8.284, SD = 5.839$), followed by *quick pulse* ($M = 4.902, SD = 3.189$), *thumb roll* ($M = 3.622, SD = 2.538$) and *baseline* ($M = 2.517, SD = 1.753$) (*table 6.4*). Nevertheless, even though *above-located targets* showed *comparable results* ($p < .05$, resp.), *differences* between *quick pulse* ($M = 3.443, SD = 2.711$) and *thumb roll* ($M = 3.289, SD = 2.470$) were not significant.

Participants were *faster* the shorter the *distance*.

Moreover, a significant *technique* × *target distance* interaction effect was found, $F(9,6065) = 10.9018, p < .0001$. For any *technique*, *participants* were *significantly faster* the *smaller* the *distance* ($p < .0001$, resp.). In this manner, *absoluteOne* performed *fastest*, followed by *small20*, *medium50* and *large80* (*table 6.4*, *figure 6.5*). Still, *natural mapping* performed *slowest* independent of *distance*, and was *significantly slower* than *quick pulse*, *thumb roll* and *baseline* ($p < .05$, resp., *table 6.4*).

Finally, there was no *evidence* to suggest that *interactions* between *menu size* and *technique* are *significant*, $F(9,6065) = 0.4361, n.s.$.

Number of Crossings

Technique had a *significant main-effect* on *number of crossings*.

Further *analysis* showed a *significant main-effect* of *technique* on *number of crossings*, $\chi^2(3) = 121.408, p < .00001$. Pairwise comparisons were performed (SPSS, 2017) with a *Bonferroni correction* for *multiple comparisons*. *Baseline* ($M = 1.3699, SD = 0.5605$) was statistically significantly different

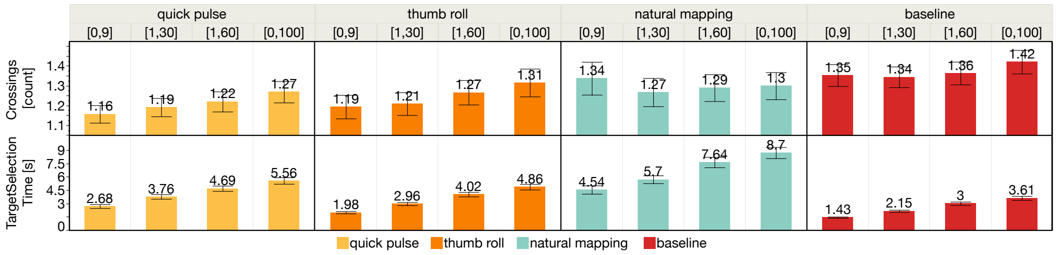


Figure 6.6: Descriptive statistics: means of *task-completion time* [*seconds*] and *number of crossings* [count data] according to **technique** and **menu size** (*error bars:* 95% CIs).

H2 is rejected.

from *quick pulse* ($M = 1.2085, SD = 0.4882$), *thumb roll* ($M = 1.2454, SD = 0.6217$) and *natural mapping* ($M = 1.298, SD = 0.7225$) (adj. $p < .0001$, resp., *figure 6.3*), and hence let to the *highest number of crossings*. As a result, returning back to the hypotheses, as introduced in *section 6.1*, it is now possible to state that we reject H_2.

Techniques showed lower *number of crossings* for *above-located targets*.

In addition, a *significant technique* × *direction* interaction effect emerged from the data, $\chi^2(7) = 254.45, p < .0001$. *Post hoc pairwise comparisons* with *Bonferroni correction* revealed that all *techniques* led to *significantly less crossings* for *above-* than *below-located targets* ($p < .005$, resp., *figure 6.4*). Moreover, independent of *direction*, *baseline* led to *significantly more crossings* than all other *techniques* (adj. $p < .05$, resp.), and showed the *highest number of crossings*, followed by *natural mapping*, *thumb roll* and *quick pulse* (*table 6.4*).

Further *examinations* found a *significant technique* × *distance* interaction effect, $\chi^2(15) = 485.167, p < .0001$. Except for *natural mapping*, all *other techniques* showed *significantly less crossings* if *small steps*, i.e., *absoluteOne*, are made (adj. $p < .05$, resp.). Only for *natural mapping*, *differences* were not *statistically significant*, $\chi^2(3) = .025, p = .999$. Equally important, *baseline* showed *significantly more crossings* than all *other techniques* if *relative distances*, i.e., *small20*, *medium50* or *large80*, are encountered (adj. $p < .005$, resp., *figure 6.5*). In contrast, when dealing with an *absolute step* of *one*, *natural mapping* ($M = 1.2958, SD = 0.7238$) showed *significantly more crossings* than *thumb roll* ($M = 1.0703, SD = 0.3653$), *quick pulse* ($M = 1.0574, SD = 0.2743$) and *baseline* ($M = 1.0208, SD = 0.143$) (adj. $p < .05$, resp., *table 6.4*).

Finally, the *analysis* also identified a *significant technique* $\times$ *menu size* interaction effect, $\chi^2(15) = 154.265, p < .0001$. Pairwise comparisons were performed (SPSS, 2017) with a *Bonferroni correction* for *multiple comparisons*. For [0,9] *menus* (*section 6.4.3*), *baseline* ($M = 1.3525, SD = 0.5497$) led to *significantly more crossings* than *thumb roll* ($M = 1.1932, SD = 0.5916$) and *quick pulse* ($M = 1.1563, SD = 0.4414$) (adj. $p < .01$, resp., *table 6.4*). Moreover, in context of [1,30], *baseline* ($M = 1.3438, SD = 0.5227$) showed *significantly higher number of crossings* than *thumb roll* ($M = 1.2083, SD = 0.572$) and *quick pulse* ($M = 1.1906, SD = 0.4607$) (adj. $p < .05$, resp., *table 6.4*). In addition, *baseline* ($M = 1.3629, SD = 0.571$) also led to *significantly more crossings* than *quick pulse* ($M = 1.2193, SD = 0.5054$) for [1,60] *menus* (adj. $p < .05$, *table 6.4*). All other *differences* were not *statistically significant*. Equally important, as illustrated in *figure 6.6*, *menu size* had no *effect* on *crossings* for any *technique*.

There was no evidence to suggest that *menu size* had an effect on *number of crossings*.

Target Accuracy

There was *strong evidence* to suggest that *technique* had a *significant main effect* on *target accuracy*, $Q(3) = 61.689, p < .0001$. Consequently, we *reject* H_3 (*section 6.1*). *McNemar's tests* with *Bonferroni correction* were applied to assess all *pairwise comparisons* (SPSS, 2017). Interestingly, *differences* between *quick pulse* (98.69%) and *thumb roll* (98.63%) were not *significant*. Still, all other *differences* were *statistically significant* ($p < .0001$, resp.). In this manner, *baseline* (100.00%) performed *most accurate*, followed by *quick pulse* (98.69%), *thumb roll* (98.63%) and *natural mapping* (96.57%) (*figure 6.3*). Please be aware that even though *quick pulse* and *thumb roll* performed *significantly less accurate* than *baseline*, the *effect* is *negligible small*, i.e., $\approx 1.37\%$.

H3 is rejected.

Further *analysis* found a *significant technique* $\times$ *direction* interaction effect, $\chi^2(7) = 121.215, p < .0001$. *Post hoc pairwise comparisons* with *Bonferroni correction* showed that *natural mapping* performed *significantly less accurate* for *down-* (94.18%) compared to *up-located targets* (98.95%) (adj. $p < .0001$). In addition, *natural mapping* was the *least accurate* if *targets* were *below* the original location (adj. $p < .0001$, resp.).

Moreover, a *significant technique*$\times$*distance* interaction effect emerged form the *data*, $\chi^2(15) = 69.156, p < .0001$. *Post hoc comparisons* with *Bonferroni correction* revealed that *baseline* was *significantly more accurate* than *natural mapping*, independent of *target distance* (adj. $p < .005$, resp.). In addition, in case of *small20*, *natural mapping* (96.32%) performed *significantly less accurate* than *quick pulse* (98.69%), *thumb roll* (99.22%) and *baseline* (100.00%) (adj. $p < .05$, resp.). Finally, *natural mapping* (95.51%) was *significantly less accurate* than *quick pulse* (98.96%) if *medium50* targets are met (adj. $p < .005$, resp.). In contrast, there was no *evidence* to suggest that *distance* had an *effect* on *target accuracy* for *any technique*.

There was no evidence to suggest that *target distance* had an effect on *target accuracy*.

Finally, the *analysis* also revealed a *significant technique*$\times$ *menu size* interaction effect $Q(15) = 78.398, p < .0001$. *Post hoc pairwise comparisons* with *Bonferroni correction* revealed that *baseline* performed *significantly more accurate* than *natural mapping*, independent of *menu size* (adj. $p < .005$, resp.). In addition, in case of [1,30], *natural mapping* (94.93%) was *significantly less accurate* than *baseline* (100.00%), *quick pulse* (99.48%) and *thumb roll* (98.44%), and hence than all other *techniques* (adj. $p < .0001$, resp.). Moreover, the *analysis* revealed that *natural mapping* (96.09%) was *significantly less accurate* than *thumb roll* (98.70%) if [1,60] *menus* are used (adj. $p < .05$). Still, there was no *evidence* to suggest that *menu size* had an *effect* on *target accuracy* for any *technique*.

Differences regarding *target accuracy* were not *significant* among menu sizes.

6.5.3 Questionnaire

Returning back to our *hypotheses*, as stated in section 6.1, H_4 requires further evaluation. Hence, *qualitative data* was analyzed using a *Friedman's test* after all *sixteen participants* completed and returned the *questionnaire* at the *end* of the *study*. As shown in *appendix B.4*, *participants* were asked to *rank techniques* from *highest*, i.e. 1, to *lowest*, i.e., 4. Please be aware that an *assignment* of the *same score* to *multiple techniques* was allowed. Interestingly, *technique* had a *significant main-effect* on *user-preference*, $\chi^2(3) = 29.25, p < .0001$. Consequently, we *reject* H_4. *Post hoc pairwise comparisons* with *Bonferroni correction* revealed that *natural mapping* ($M = 3.63, SD = 0.50$) was *significantly less preferred* than all

Participants were asked to state their *personal-preference* regarding all techniques from *highest* to *lowest*.

H4 is rejected.

other techniques ($p < .05$, resp.). However, *differences* between *quick pulse* ($M = 2.06, SD = 0.68$), *thumb roll* ($M = 1.88, SD = 0.89$) and *baseline* ($M = 1.56, SD = 0.90$) were not significant. Having referred to *findings* with respect to *quantitative* as well as *qualitative data*, we briefly *discuss* these results before drawing a *conclusion* at the end of the *chapter*.

6.5.4 Discussion

The most striking *result* that emerged from the data is that *thumb roll* and *quick pulse* represent *appropriate solutions* to the *bidirectional problem*, and hence enable *bidirectional force input* from a *static location*. Even though *performance* of *force techniques* in terms of *completion time* was not ideal, *accuracy* for *thumb roll* and *quick pulse* was remarkably *high*, i.e., almost 99%. Hence, *participants* could successfully *acquire* and *select* the intended *target* using both *pressure-based interaction designs* that justifies their *applicability* in practice.

Thumb roll and *quick pulse* represent *appropriate solutions* to the *bidirectional problem*.

Equally important, *both techniques* could outperform *baseline* in terms of *number of crossings*. A possible *explanation* is given by the *observation* that *participants* were *less afraid* of doing *over-* or *undershoots* in the *baseline condition*. In addition, we expected *baseline* to outperform *remaining designs* in terms of *preference* and *completion time*, since *users* were *well-familiar* with *multitouch-input*. With this in mind, we believe that even though *thumb roll* performed $\approx 910ms$ slower than *baseline*, *differences* can be minimized with *further practice*.

We are confident that *users* can become *faster* with further *training*.

Clearly, *natural mapping* can not be considered as an *appropriate solution*, since it performed *slowest* and was the *least preferred*. In this regard, *user feedback* revealed that *participants* had issues *controlling force* during *pressure release*, and hence took *significantly longer* to finalize their *selection*. Interestingly, these *findings* are consistent with *previous results* that identified *usability issues* of *pressure pattern* in case of *positional control* (*section 5.6*). Therefore, against our *expectations* that *performance* would improve with *rate-based control*, there is *evidence* to suggest that an *inverted mapping*, i.e., assigning *pressure-release* to *value-decrease*, causes *confusion* and is *too mentally demanding*.

Natural mapping is not an *appropriate solution*.

Moreover, this *result* is justified by *performance data* of the *small-step condition* that revealed that *natural mapping* performed *sig. slower* and led to *sig. more crossings* than all other *techniques*. *Possible reasons* include the *finiteness* of the *force-sensitive range*. In this regard, *participants* experienced *usability issues* when making *small-step adjustments*, since they could not estimate when *actions* will have an effect. This is because *visual feedback* can not be provided when the *maximum* of the *force-sensitive range* is exceeded. Hence, *users* released *pressure* too *quickly*, causing *inadvertent overshoots* that affected the *technique's accuracy*. Still, these *results* did not apply to *thumb roll* and *quick pulse*, since *small-steps* are *easily made* through slight *force variations*.

Participants had issues making *small-step adjustments* with *natural mapping*.

Above-stated findings also revealed that *force techniques* performed *significantly faster* for *above-* than *below-located targets*. This *result* was expected, since the *initial direction* was always set to *increase*. Hence, navigating in the *opposite direction* first required *participants* to *specify directions* causing *inevitable delays*. However, even though differences were *significant*, *thumb roll* showed the *smallest difference* of $0.33s$, compared to $1.46s$ by *quick pulse* and $3.25s$ by *natural mapping*. This is because *thumb roll* offers *immediate access* to *both directions* and hence does not suffer from *artificial delays*.

Performance differences in terms of *completion time* were lowest for *thumb roll*, since it offers immediate access to *both directions*.

6.6 Conclusion

Taken together, this *chapter* has concentrated on the *second research question* to identify the *technique* that *performs best* and is *most preferred*. In this regard, it is now possible to state that *thumb roll* and *quick pulse* provide an *appropriate solution* to the *bidirectional problem*, since *users* could perform *bidirectional value manipulations* with *high accuracy* of almost 99%. On the contrary, *natural mapping* is omitted, since it performed *worst* and was the *least* preferred. Even though *baseline* performed *faster* than *thumb roll* and *quick pulse*, *differences* in *preference* were not *significant*. In addition, *users* noted that they were *more cautious* with *force techniques*, since they were not as *familiar* as *baseline*. Consequently, we are confident that *thumb roll* and *quick pulse* become *faster* with *further practice*.

Thumb roll and *quick pulse* allow *bidirectional value manipulations* with *high accuracy*.

Chapter 7

Summary and Future Work

7.1 Summary and Contributions

With the introduction of *force-sensing capabilities* to *mobile devices* like *smartphones*, *pressure-based input* has become available to many *people*. Nevertheless, *current applications* have not yet taken *advantage* of the *full potential* that *force input* can bring to the *field*. In this regard, *force* can supplement *multitouch input* by offering an *additional dimension* that assigns *enriched functionality* to the *same limited space*. As a result, *users* can stay within their *comfortable interaction range* and do not require *significant changes* in *hand posture*, since *force* is controlled from a *static location*. These *characteristics* are *beneficial* to counteract any *adverse effects* caused by *reachability-* or *occlusion issues* that typically arise when operating *smartphones single-handed* and only using the *thumb*.

Force offers an *additional dimension* compared to *multi-touch input*.

However, *force input* suffers *constraints* posed by the *bidirectional problem*. In this regard, even though *recent work* in the *area* has demonstrated *several examples* of *force input* in context of *mouse-*, *pen-* and *multitouch-interaction*, the majority of these *applications* are *unidirectional*. Still, most *controls* require *value manipulations* in *both directions* such that the *main objective* of this *thesis* was to tackle the *bidirectional problem*.

In this *thesis* we have followed a *systematic procedure* according to our *research questions*, since there was no *trivial solution* to the best of our knowledge. In this regard, we first identified *three essential components* that are required to enable *bidirectional force input* from a *static location* and took a *glimpse* at the *design space* of *bidirectional force input*, from which *eighteen designs* were derived.

Findings of the *first study* suggested to focus on *three remaining designs*, namely *quick pulse*, *thumb roll* and *natural mapping*.

In the *first study* we investigated which combination of *pressure mapping*, *direction-* and *pressure control mechanism performs best* and is *most preferred* to only focus on *designs* that are built from *remaining components*. *Findings* led us to conclude that *[1:1]-mappings* should be omitted from *further considerations*, since *users* are still *fast* when always starting from *zero-force*, and hence do not need to *specify directions*. In addition, we decided to consider *rate-based-* rather than *positional control*, since *positional pumping* was *slow* and resulted in *higher number of crossings*. Consequently, we focussed on *three remaining designs*, namely *quick pulse*, *thumb roll* and *natural mapping* that utilize *rate-based control* within *multiple regions*. Note that *quick pulse* was obtained by merging *maximum force* and *double pulse* respectively, while *natural mapping* depicted *pressure pattern* with *rate-based control*.

Thumb roll and *quick pulse* were liked by *participants*.

Based on these *results*, we have finally drawn our attention to the *second research question* and compared *remaining designs* in terms of *user-preference* and *performance* against a *baseline condition*. Findings revealed that *thumb roll* and *quick pulse* represent *appropriate solutions* to the *bidirectional problem*, since *users* could perform *bidirectional value manipulations* with *great accuracy* of almost 99%. Even though *baseline* performed $\approx 910ms$ *faster* than *thumb roll*, these results were expected, since *users* were already familiar with *multitouch input*. Hence, since *thumb roll* and *quick pulse* were overall liked by *participants* and achieved great *accuracy*, we are encouraged that *performance differences* in terms of *completion time* are minimized with *further training*.

This *thesis* has provided the *first step* to counteract the *main limitation* of *force*, i.e., the *bidirectional problem*. *Results* are *beneficial* to *interaction designers* to take advantage of *human's profound force-sensing capabilities* and make *force input* applicable in *additional domains*. Still, *work* needs to be done.

7.2 Limitations and Future Work

Finally, some *limitations* need to be considered. *First*, our *investigations* only aimed for *right-handed participants*, since an *unbalanced distribution* of *handedness* would have confounded our *results*. Nevertheless, *presented designs* should also *work* for *left-handed participants* if *directions* are swapped for *thumb roll*. As a result, even though we expect *results* to be *symmetric*, this *hypothesis* needs to be *tested*.

We only considered *right-handed participants*.

In addition, *presented techniques* are meant to be used within *ordered domains*, e.g., *number lists*, since they provide *necessary context information* to estimate where an *intended value* is found. Nevertheless, note that *bidirectional designs* also *adapt* to *similar domains* like *temperature*, *brightness* or *volume-control*, and also work in context of *alphabetical ordered items*, like *music-* or *contact-lists*.

Proposed bidirectional designs are meant to be used within *ordered-domains*.

Moreover, our *research* only focussed on *discrete-* rather than *continuous selections*. Still, please be aware that *input* provided by *force-techniques* is *continuous* and hence can be easily mapped to *additional controls*, like *scroll views* or *sliders*. In this regard we already utilized *continuous input* to provide *visual feedback* about the *picker's content location*. Hence, we are confident that *bidirectional designs* will show *similar performance* when *choosing values* out of *continuous domains*.

We only examined *discrete value selection*.

Unfortunately, we could not *evaluate* our *designs* under *different environmental conditions* due to an *increasing number of conditions*. That's why *participants* only performed the *target acquisition* and *selection task* while *sitting*. Having identified *thumb roll* and *quick pulse* as *appropriate solutions*, *future work* should *evaluate* both *techniques* while *standing* and *walking*.

Bidirectional designs were only tested while *sitting*.

Equally important, the *poor performance* of *natural mapping* for both *positional-* as well as *rate-based control* led us to conclude that *controlling force* during *pressure release* is more difficult than *navigating* form *zero-force*. Nevertheless, this *hypothesis* needs to be *confirmed* by an *empirical evaluation*.

Finally, we draw the *reader's attention* to an *interesting observation*, derived from *continuous data* obtained in the *study*.

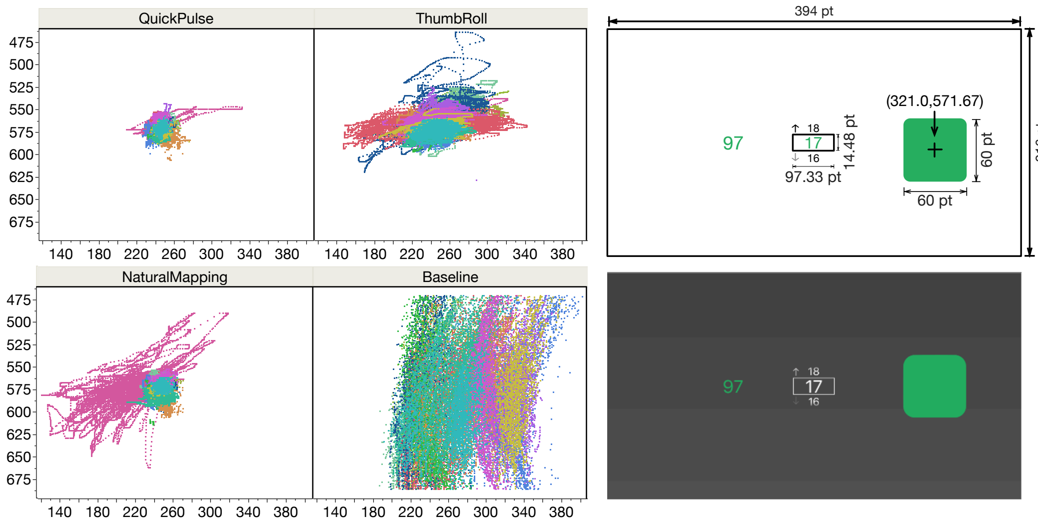


Figure 7.1: Left: *gesture footprint* of *quick pulse*, *thumb roll*, *natural mapping* and *baseline* respectively. *Touch locations* are measured in *points*, colored according to the *participant ID*, Right: *smaller picker* including *dimensions*, measured in *points*.

The *gesture footprint* of *force techniques* is *significantly smaller* compared to *baseline*.

In this regard, we logged the *user's touch location*, *force* and *current selected value* every $16ms$. *Figure 7.1, left* visualizes the *gesture footprint* that is defined by the *touch locations* of all *participants* throughout the *study*. Clearly, the *footprint* of *force techniques* is *significantly smaller* compared to *baseline*. In addition, note that *widgets* implementing *bidirectional designs* only require that the *initial contact position* is located within the *widgets' space*. Consequently, *users* are allowed to *drift* from the *widget's location*, resulting in *smaller screen-space requirements*. As a result, this *observation* suggests that *bidirectional designs* can not only overcome *reachability*- and *occlusion issues* within *one-handed use*, but also enable *space-efficient input* using the *third dimension* offered by *force*.

Our bidirectional designs seem to be *well-suited* for *space-efficient input*, since their *gesture footprint* is *small*.

We are currently *investigating* this *observation*, and have already come up with a *smaller picker representation*, as illustrated in *figure 7.1, right*. In this regard, the *picker* only measures $97.33pt \times 43.44pt$ in size and uses a $17.0pt$ *font*. As a result, *users* can perform *bidirectional value manipulations* using *thumb roll* or *quick pulse* alongside the *picker*. Possible *applications* include *space-efficient in-row selections* that do not rely on a *standard-sized picker*, and hence offer *more space* for *content*. Still, an *evaluation* is beyond the *scope* of this *thesis*.

Appendix A

Consent Forms

Subsequently, *consent forms* for *both studies* are stated. In this regard, *participants* were asked for their *written approval* that *measurement data* may be used for the *statistical evaluation*. Indeed, *responses* were *completely anonymized* and only *associated* with the *participant ID*. In addition, *participants* were informed that they might become *fatigue* while *performing* the *target-acquisition* and *selection-task* throughout the *study*. Nevertheless, *breaks* were offered whenever needed.

Participant's responses were completely annonymized.

Informed Consent Form

Bidirectional Force Input

PRINCIPAL INVESTIGATOR Andreas Link.
Media Computing Group
RWTH Aachen University
Phone: +4915237715179
Email: andreas.link@rwth-aachen.de

Context: Pressure has been proposed as additional input channel to supplement conventional interaction modalities as used in the desktop-, tablet- or smartphone- domain. This way, pressure does not require significant changes in hand posture and can be controlled from a static location. These characteristics are beneficial to counteract any adverse effects caused by *visual occlusion* or *reachability issues*. However, *pressure-control* is limited in the way that it is *unidirectional*. To take on this issue, we have proposed several *interaction designs* that permit *bidirectional force input* from a static location.

Purpose of the study: The purpose of this study is to get initial insights about which combination of the *three essential* components, namely *pressure* mapping, *direction-*, as well as *pressure-control* mechanism, *performs best* and is *most* preferred by participants. Findings allow us to remove the *combinations* that should better be excluded and evaluate *remaining designs* by comparing their performance against a *baseline condition*.

Procedure: In this study you will perform *target acquisition and selection tasks* while operating a smartphone single-handed and only using your thumb. Users are allowed to get familiar with the proposed *bidirectional designs* during an *initial test run*. Throughout the study you will be asked to fill in sections of a questionnaire.

Risks/Discomfort: You may become fatigued during the study. You will be given several opportunities to rest, and additional breaks are also possible. There are no other risks associated with participation in the study. Should completion of either the task or the questionnaire become distressing to you, it will be terminated immediately.

Alternatives to Participation: Participation in this study is voluntary. You are free to withdraw or discontinue the participation.

Cost and Compensation: Participation in this study will involve no cost to you. There will be sweets offered during and after participation.

Confidentiality: All information collected during the study period will be kept strictly confidential. You will be identified through identification numbers. No publications or reports from this project will include identifying information on any participant. If you agree to join this study, please sign your name below.

___ I have read and understood the information on this form.
___ I have had the information on this form explained to me.

Participant's Name	Participant's Signature	Date
	Principal Investigator	Date

If you have any questions regarding this study, please contact Andreas Link at +49152 37715179, email: andreas.link@rwth-aachen.de

Figure A.1: *Consent form*, as used in the *first study*.

Informed Consent Form

Bidirectional Force Input

PRINCIPAL INVESTIGATOR Andreas Link.
Media Computing Group
RWTH Aachen University
Phone: +4915237715179
Email: andreas.link@rwth-aachen.de

Context: Pressure has been proposed as additional input channel to supplement conventional interaction modalities as used in the desktop-, tablet- or smartphone- domain. This way, pressure does not require significant changes in hand posture and can be controlled from a static location. These characteristics are beneficial to counteract any adverse effects caused by *visual occlusion* or *reachability issues*. However, *pressure-control* is limited in the way that it is *unidirectional*. To take on this issue, we have proposed several *interaction designs* that permit *bidirectional force input* from a static location.

Purpose of the study: The purpose of this study is to investigate the applicability of three bidirectional designs, namely **Quick Pulse**, **Thumb Roll** and **Natural Mapping** by comparing their *performance* against a **Baseline condition**. Findings allow us to identify the *design* that is *most preferred* and *performs best.*

Procedure: In this study you will perform *target acquisition and selection tasks* while operating a smartphone single-handed and only using your thumb. Users are allowed to get familiar with the proposed *bidirectional designs* during *initial testing*. Throughout the study you will be asked to fill in sections of a questionnaire.

Risks/Discomfort: You may become fatigued during the study. You will be given several opportunities to rest, and additional breaks are also possible. There are no other risks associated with participation in the study. Should completion of either the task or the questionnaire become distressing to you, it will be terminated immediately.

Alternatives to Participation: Participation in this study is voluntary. You are free to withdraw or discontinue the participation.

Cost and Compensation: Participation in this study will involve no cost to you. There will be sweets offered during and after participation.

Confidentiality: All information collected during the study period will be kept strictly confidential. You will be identified through identification numbers. No publications or reports from this project will include identifying information on any participant. If you agree to join this study, please sign your name below.

___ I have read and understood the information on this form.
___ I have had the information on this form explained to me.

Participant's Name	Participant's Signature	Date
	Principal Investigator	Date

If you have any questions regarding this study, please contact Andreas Link at +49152 37715179, email: andreas.link@rwth-aachen.de

Figure A.2: *Consent form*, as used in the *second study*.

Appendix B

Questionnaires

To address the *second research question* (*section 2.6*), we also collected *preference-* in addition to *quantitative data*. Hence, *questionnaires* for *both studies* are stated. In the *first study*, *participants* were asked to judge their *preference* on a *seven-point likert scale*, i.e., from *totally disagree* to *totally agree*, for each *bidirectional design* respectively (*figure B.1*, *figure B.2*). In addition, *user's overall preference* regarding *direction-* and *pressure-control mechanisms* was assessed through *rankings* from *highest*, i.e., 1, to *lowest*, i.e., 5. Indeed, *participants* could assign the *same score* to multiple *components*.

Participants were allowed to assign the *same score* to *multiple components*.

In contrast, the *questionnaire* from the *second study* focussed on *four criteria*, namely *controllability*, *speed*, *overshoot-* and *undershoot-corrections* that were judged on a *seven-point likert scale* (*figure B.3*, *figure B.4*). In this regard, we assessed *user-preference* for each of the *remaining designs*. Finally, *participants* were asked to rank all *techniques* (incl. *baseline*) from *highest* to *lowest*.

QUESTIONNAIRE

Participant ID: _______

(1) Gender: ◯ Male ◯ Female

(2) Age: _______________

(3) Please rate your **level of agreement** on the following **statements:**

Block 1:

	Target acquisition and **selection** was **easy** to perform using:	totally disagree			neither			totally agree
T1	One-to-one **max-force** positional navigation	☐	☐	☐	☐	☐	☐	☐
T2	One-to-one **thumb-bob** positional navigation	☐	☐	☐	☐	☐	☐	☐
T3	One-to-one **thumb-roll** positional navigation	☐	☐	☐	☐	☐	☐	☐
T4	One-to-one **double-pulse** positional navigation	☐	☐	☐	☐	☐	☐	☐
T5	One-to-one **pressure-pattern** positional navigation	☐	☐	☐	☐	☐	☐	☐

Block 2:

	Target acquisition and **selection** was **easy** to perform using:	totally disagree			neither			totally agree
T6	One-to-one **max-force** rate-based control	☐	☐	☐	☐	☐	☐	☐
T7	One-to-one **thumb-bob** rate-based control	☐	☐	☐	☐	☐	☐	☐
T8	One-to-one **thumb-roll** rate-based control	☐	☐	☐	☐	☐	☐	☐
T9	One-to-one **double-pulse** rate-based control	☐	☐	☐	☐	☐	☐	☐

Block 3:

	Target acquisition and **selection** was **easy** to perform using:	totally disagree			neither			totally agree
T10	One-to-many **max-force** positional pumping	☐	☐	☐	☐	☐	☐	☐
T11	One-to-many **thumb-bob** positional pumping	☐	☐	☐	☐	☐	☐	☐
T12	One-to-many **thumb-roll** positional pumping	☐	☐	☐	☐	☐	☐	☐
T13	One-to-many **double-pulse** positional pumping	☐	☐	☐	☐	☐	☐	☐
T14	One-to-many **pressure-pattern** positional pumping	☐	☐	☐	☐	☐	☐	☐

Figure B.1: *Questionnaire's front*, as used in the *first study*.

Block 4:

	Target acquisition and **selection** was **easy** to perform using:	totally disagree			neither			totally agree
T15	One-to-many **max-force** rate-based control	☐	☐	☐	☐	☐	☐	☐
T16	One-to-many **thumb-bob** rate-based control	☐	☐	☐	☐	☐	☐	☐
T17	One-to-many **thumb-roll** rate-based control	☐	☐	☐	☐	☐	☐	☐
T18	One-to-many **double-pulse** rate-based control	☐	☐	☐	☐	☐	☐	☐

(4) Please **rank** the following **direction mechanisms** according to your **personal preference:**

		Ranking: (1 = highest, 5 = lowest)
D1	**Maximum-force**	
D2	**Thumb-bob**	
D3	**Thumb-roll**	
D4	**Double-pulse**	
D5	**Pressure-Pattern**	

(5) Please **rank** the following **pressure control mechanisms** according to your **personal preference:**

		Ranking: (1 = highest, 2 = lowest)
P1	**Positional Control**	
P2	**Rate-based Control**	

Figure B.2: *Questionnaire's back*, as used in the *first study*.

Questionnaire (Fragebogen)

Participant ID (Teilnehmer Nummer): _______

(1) Gender (Geschlecht): ◯ female (weiblich) ◯ male (männlich)

(2) Age (Alter): _______________

(3) Please rate your level of agreement on the following statements
(Bitte beurteilen Sie Ihre Zustimmung zu den folgenden Aussagen)

Technique 1: **Quick Pulse** (Schneller Impuls)

	Statements:	totally disagree (lehne stark ab)			neither (weder-noch)			totally agree (stimme stark zu)
S1	**Controllability:** Target acquisition and selection was easy to perform. **Kontrollierbarkeit:** Die Zielerfassung und Auswahl war einfach durchzuführen.	☐	☐	☐	☐	☐	☐	☐
S2	**Speed:** Trials could be quickly completed. **Geschwindigkeit:** Versuche konnten schnell abgeschlossen werden.	☐	☐	☐	☐	☐	☐	☐
S3	**Overshoot-corrections** using small steps could be easily made. **Zielüberschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐
S4	**Undershoot-corrections** using small steps could be easily made. **Zielunterschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐

Technique 2: **Thumb Roll** (Daumen Rolle)

	Statements:	totally disagree (lehne stark ab)			neither (weder-noch)			totally agree (stimme stark zu)
S1	**Controllability:** Target acquisition and selection was easy to perform. **Kontrollierbarkeit:** Die Zielerfassung und Auswahl war einfach durchzuführen.	☐	☐	☐	☐	☐	☐	☐
S2	**Speed:** Trials could be quickly completed. **Geschwindigkeit:** Versuche konnten schnell abgeschlossen werden.	☐	☐	☐	☐	☐	☐	☐
S3	**Overshoot-corrections** using small steps could be easily made. **Zielüberschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐
S4	**Undershoot-corrections** using small steps could be easily made. **Zielunterschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐

Technique 3: **Natural Mapping** (Natürliche Zuordnung)

	Statements:	totally disagree (lehne stark ab)			neither (weder-noch)			totally agree (stimme stark zu)
S1	**Controllability:** Target acquisition and selection was easy to perform. **Kontrollierbarkeit:** Die Zielerfassung und Auswahl war einfach durchzuführen.	☐	☐	☐	☐	☐	☐	☐
S2	**Speed:** Trials could be quickly completed. **Geschwindigkeit:** Versuche konnten schnell abgeschlossen werden.	☐	☐	☐	☐	☐	☐	☐
S3	**Overshoot-corrections** using small steps could be easily made. **Zielüberschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐
S4	**Undershoot-corrections** using small steps could be easily made. **Zielunterschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐

Figure B.3: *Questionnaire's front*, as used in the *second study*.

Technique 4: **Baseline** (Vergleichskondition)

	Statements:	totally disagree (lehne stark ab)			neither (weder-noch)			totally agree (stimme stark zu)
S1	**Controllability:** Target acquisition and selection was easy to perform. **Kontrollierbarkeit:** Die Zielerfassung und Auswahl war einfach durchzuführen.	☐	☐	☐	☐	☐	☐	☐
S2	**Speed:** Trials could be quickly completed. **Geschwindigkeit:** Versuche konnten schnell abgeschlossen werden.	☐	☐	☐	☐	☐	☐	☐
S3	**Overshoot-corrections** using small steps could be easily made. **Zielüberschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐
S4	**Undershoot-corrections** using small steps could be easily made. **Zielunterschreitungen** konnten leicht mittels kleiner Schritte korrigiert werden.	☐	☐	☐	☐	☐	☐	☐

(4) Please rank the following techniques according to your personal preference
(Bitte ordnen Sie die folgenden Techniken nach Ihrer persönlichen Präferenz)

		Ranking: (1 = highest/höchstes, 4 = lowest/niedrigstes)
T1	Quick Pulse (Schneller Impuls)	
T2	Thumb Roll (Daumen Rolle)	
T3	Natural Mapping (Natürliche Zuordnung)	
T4	Baseline (Vergleichskondition)	

(6) Please leave comments or suggestions down below
(Bitte hinterlassen Sie Kommentare oder Vorschläge im unteren Feld)

Figure B.4: *Questionnaire's back*, as used in the *second study*.

Appendix C

Force Profiles

In addition to *responses*, like *task-completion time*, *number of crossings* and *target accuracy*, *continuous input data* was logged throughout the *study*. These *measurements* allowed us to create *force profiles*, as suggested by Taher et al. [2014], that illustrate the *shape* of *value-* and *force-curve* over *time*. In this regard, *figure C.1 (A)* corresponds to the *task* to navigate from $7 \rightarrow 19$ using *thumb roll* as *bidirectional force technique*. Interestingly, the *value* increases from $0.0s$ to $0.875s$ until an *overshoot* has occurred. Consequently, the *current selected value* is adjusted by *rolling* in the opposite direction. Note that a *pumping pattern* is used to allow *small steps* until the *intended target* is accomplished (*figure C.1, A*).

Force profiles illustrate *value-* and *force-curve* over time.

In contrast, *figure C.1 (B)* visualizes *quick pulse* to navigate from $60 \rightarrow 12$. Note that the *direction* is *toggled* at the *beginning* of the *trial* such that the *value decreases* if *force* is applied. Equally important, participants *decelerated* while *closing in* on the *target* such that *overshoots* are avoided.

Participants slowed down when *closing in* on the *target*.

Finally, the *force profile* of *natural mapping* when navigating from $100 \rightarrow 20$ is illustrated in *figure C.1 (C)*. In this manner, *force* is *quickly applied*, followed by slow *pressure release*. Indeed, *pressure application* is required to *decelerate* before reaching the *target*. Interestingly, *natural mapping* seems to be *less accurate*, since the *force-curve* suffers from *strong fluctuations*.

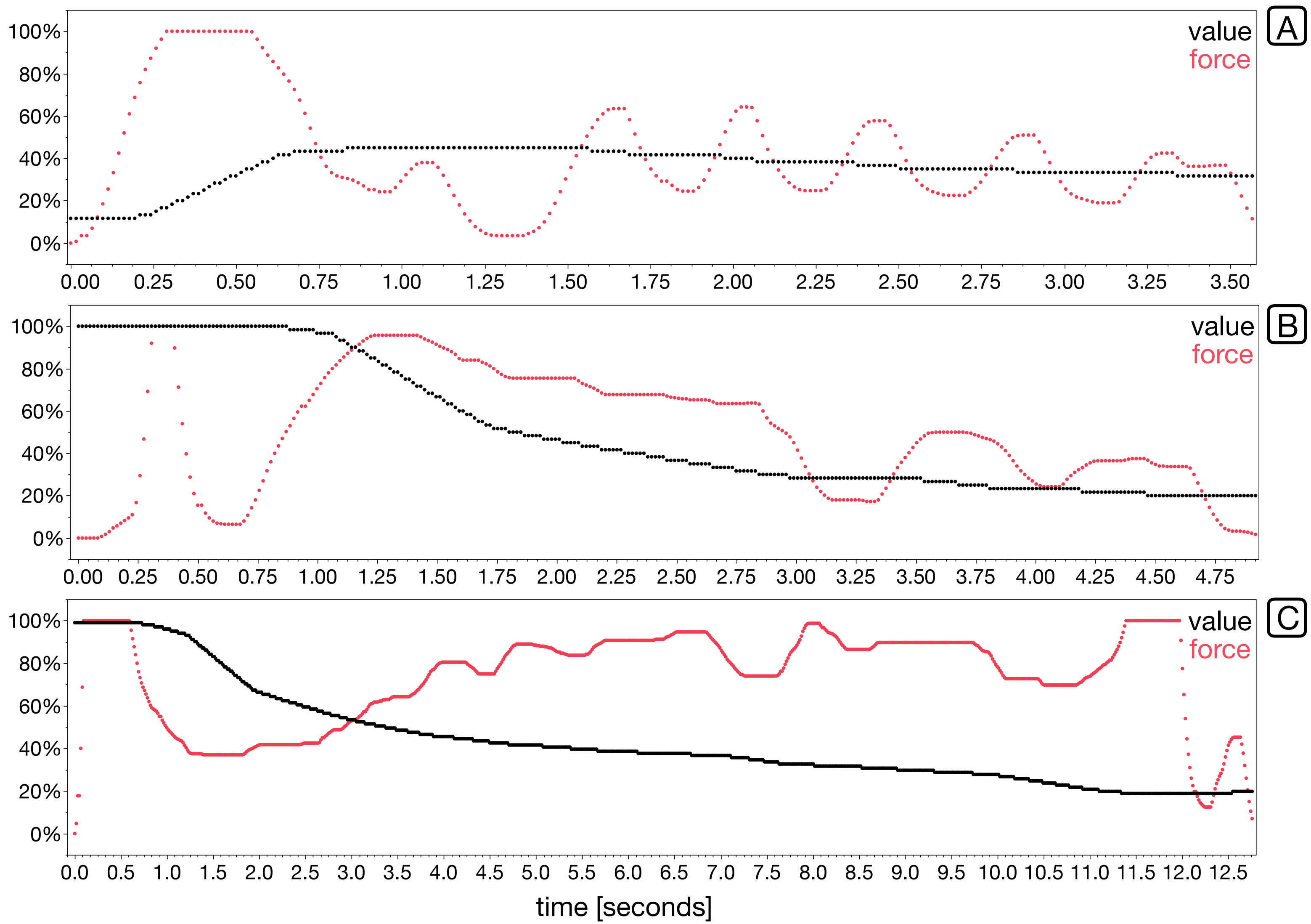


Figure C.1: Force Profiles (*Study 2*): *value* (*black*) and *force* (*red*) normalized over *time*. A: *thumb-roll* (7→19), B: *quick pulse* (60→12), C: *natural mapping* (100→20).

Bibliography

3D4Medical. Hand anatomy. https://3d4medical.com/, 2017. Accessed: July 15, 2017.

Adam and Mark Lund. Anova assumptions. https://statistics.laerd.com/spss-tutorials/one-way-anova-using-spss-statistics.php, 2017. Accessed: September 7, 2017.

Youngseok Ahn, Sungjae Hwang, HyunGook Yoon, Junghyeon Gim, and Jung-hee Ryu. BandSense: Pressure-sensitive multi-touch interaction on a wristband. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, pages 251–254, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3146-3. doi: 10.1145/2702613.2725441. URL http://doi.acm.org/10.1145/2702613.2725441.

Axel Antoine, Sylvain Malacria, and Géry Casiez. Forceedge: Controlling autoscroll on both desktop and mobile computers using the force. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3281–3292, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025605. URL http://doi.acm.org/10.1145/3025453.3025605.

Apple©. Airdrop. https://support.apple.com/en-us/HT204144, 2017a. Accessed: September 4, 2017.

Apple©. GKStateMachine. https://developer.apple.com/documentation/gameplaykit/gkstatemachine, 2017b. Accessed: August 29, 2017.

Apple©. Adopting 3D touch on iPhone. https://developer.apple.com/library/content/documentation/UserExperience/Conceptual/Adopting3DTouchOniPhone/, 2017c. Accessed: August 14, 2017.

Apple©. UIKit. https://developer.apple.com/documentation/uikit, 2017d. Accessed: August 28, 2017.

Apple©. iPhone 6s Plus technical specification. https://www.apple.com/iphone-6s/specs/, 2017e. Accessed: May 10, 2017.

Gábor Blaskó and Steven Feiner. Single-handed interaction techniques for multiple pressure-sensitive strips. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '04, pages 1461–1464, New York, NY, USA, 2004. ACM. ISBN 1-58113-703-6. doi: 10.1145/985921.986090. URL http://doi.acm.org/10.1145/985921.986090.

Sebastian Boring, David Ledo, Xiang 'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. The fat thumb: Using the thumb's contact size for single-handed mobile interaction. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '12, pages 39–48, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1105-2. doi: 10.1145/2371574.2371582. URL http://doi.acm.org/10.1145/2371574.2371582.

William Buxton, Ralph Hill, and Peter Rowley. Issues and techniques in touch-sensitive tablet input. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '85, pages 215–224, New York, NY, USA, 1985. ACM. ISBN 0-89791-166-0. doi: 10.1145/325334.325239. URL http://doi.acm.org/10.1145/325334.325239.

California State University. Wrist and hand. https://www.studyblue.com/notes/note/n/an1-08-pic-wrist-and-hand/deck/8890050, 2017. Accessed: August 12, 2017.

A. Campos, C. F. Neto, M. Cansian, A. Reis, and N. Borges. Touchscreen device layout based on thumb comfort and precision. In *5th IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*, pages 975–980, Aug 2014. doi: 10.1109/BIOROB.2014.6913907.

Jared Cechanowicz, Pourang Irani, and Sriram Subramanian. Augmenting the mouse with pressure sensitive input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1385–1394, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9. doi: 10.1145/1240624.1240835. URL `http://doi.acm.org/10.1145/1240624.1240835`.

JV Chamary. 3D Touch In iPhone 6S Isn't Just A Gimmick. Here's How It Works. `https://www.forbes.com/sites/jvchamary/2015/09/12/3d-touch-iphone-6s/#1bbf8d634cee`, 2015. Accessed: May 10, 2017.

R. Chang, F. Wang, and P. You. A survey on the development of multi-touch technology. In *2010 Asia-Pacific Conference on Wearable Computing Systems*, pages 363–366, April 2010. doi: 10.1109/APWCS.2010.99.

Chicago Public Media. This American Life (podcast). `https://www.thisamericanlife.org`, 2017. Accessed: July 14, 2017.

David Cilia. CSV-Export Swift. `https://github.com/dcilia/csv-export-swift`, 2017. Accessed: September 4, 2017.

Rajkumar Darbar, Prasanta Kr Sen, and Debasis Samanta. Presstact: Side pressure-based input for smartwatch interaction. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '16, pages 2431–2438, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4082-3. doi: 10.1145/2851581.2892436. URL `http://doi.acm.org/10.1145/2851581.2892436`.

Staas de Jong, Dünya Kirkali, Hanna Schraffenberger, Jeroen Jillissen, Alwin de Rooij, and Arnout Terpstra. One-press control: A tactile input method for pressure-sensitive computer keyboards. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, pages 4261–4266, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-930-5. doi: 10.1145/1753846.1754136. URL http://doi.acm.org/10.1145/1753846.1754136.

Shimin Feng, Graham Wilson, Alex Ng, and Stephen Brewster. Investigating pressure-based interactions with mobile phones while walking and encumbered. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, MobileHCI '15, pages 854–861, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3653-6. doi: 10.1145/2786567.2793711. URL http://doi.acm.org/10.1145/2786567.2793711.

FlatIcon. Speedometer. https://www.flaticon.com/free-icon/speedometer_427111, 2017. Accessed: August 23, 2017.

Clifton Forlines, Chia Shen, and Bill Buxton. Glimpse: A novel input model for multi-level devices. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pages 1375–1378, New York, NY, USA, 2005. ACM. ISBN 1-59593-002-7. doi: 10.1145/1056808.1056920. URL http://doi.acm.org/10.1145/1056808.1056920.

Masitah Ghazali and Alan Dix. Knowledge of today for the design of tomorrow. In *in Proceedings of the 2nd International Design and Engagibility Conference (IDEC*, pages 2–7, 2005.

Douglas C Giancoli. *Physics: principles with applications*. Pearson Education USA, 2005.

Aurora Harley. Slider Design: Rules of Thumb. https://www.nngroup.com/articles/gui-slider-controls/, 2015. Accessed: September 13, 2017.

Fabian Hemmert, Matthias Löwe, Anne Wohlauf, and Gesche Joost. Tactful calling: Urgency-augmented phone calls through high-resolution pressure input on mobile phones. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 3187–3192, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-247-4. doi: 10.1145/1520340.1520454. URL `http://doi.acm.org/10.1145/1520340.1520454`.

Seongkook Heo and Geehyuk Lee. ForceDrag: Using pressure as a touch input modifier. In *Proceedings of the 24th Australian Computer-Human Interaction Conference*, OzCHI '12, pages 204–207, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1438-1. doi: 10.1145/2414536.2414572. URL `http://doi.acm.org/10.1145/2414536.2414572`.

Christopher F. Herot and Guy Weinzapfel. One-point touch input of vector information for computer displays. In *Proceedings of the 5th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '78, pages 210–216, New York, NY, USA, 1978. ACM. doi: 10.1145/800248.807392. URL `http://doi.acm.org/10.1145/800248.807392`.

Nambu Hirotaka. Reassessing current cell phone designs: Using thumb input effectively. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '03, pages 938–939, New York, NY, USA, 2003. ACM. ISBN 1-58113-637-4. doi: 10.1145/765891.766081. URL `http://doi.acm.org/10.1145/765891.766081`.

Jochen Huber, Mohamed Sheik-Nainar, and Nada Matic. Towards an interaction language for force-enabled touchpads in cars. In *Adjunct Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '16 Adjunct, pages 197–202, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4654-2. doi: 10.1145/3004323.3004347. URL `http://doi.acm.org/10.1145/3004323.3004347`.

Scott Hurff. How to design for thumbs in the era of huge screens. `http://scotthurff.com/posts/how-to-design-for-thumbs-in-the-era-of-huge-screens`, 2017. Accessed: July 13, 2017.

IOSTE. Map. `http://www.eventos.ciec-uminho.org/ioste2016/images/ioste/MAP.png`, 2016. Accessed: August 23, 2017.

Roland S Johansson and J Randall Flanagan. Coding and use of tactile signals from the fingertips in object manipulation tasks. *Nature reviews. Neuroscience*, 10(5):345, 2009.

Amy K. Karlson and Benjamin B. Bederson. Studies in one-handed mobile design: Habit, desire and agility. Technical report, Proceedings of the 4th ERCIM Workshop on User Interfaces for All (UI4ALL '98), 2006a.

Amy K. Karlson and Benjamin B. Bederson. Understanding single-handed mobile device interaction. Technical report, 2006b.

Amy K. Karlson and Benjamin B. Bederson. Thumbspace: Generalized one-handed input for touchscreen-based mobile devices. In *Proceedings of the 11th IFIP TC 13 International Conference on Human-computer Interaction*, INTERACT'07, pages 324–338, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 3-540-74794-X, 978-3-540-74794-9. URL `http://dl.acm.org/citation.cfm?id=1776994.1777034`.

Matthew Kay. Artool. `https://github.com/mjskay/ARTool`, 2017. Accessed: September 6, 2017.

I. Scott MacKenzie. Movement time prediction in human-computer interfaces. pages 483–492. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995. ISBN 1-55860-246-1. URL `http://dl.acm.org/citation.cfm?id=212925.212969`.

Dinesh Mandalapu and Sriram Subramanian. Exploring pressure as an alternative to multi-touch based interaction. In *Proceedings of the 3rd International Conference on Human Computer Interaction*, IndiaHCI '11, pages 88–92, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0729-1. doi: 10.1145/2407796.2407810. URL `http://doi.acm.org/10.1145/2407796.2407810`.

David C. McCallum, Edward Mak, Pourang Irani, and Sriram Subramanian. Pressuretext: Pressure input for mobile phone text entry. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 4519–4524, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-247-4. doi: 10.1145/1520340.1520693. URL `http://doi.acm.org/10.1145/1520340.1520693`.

Ross McLachlan, Daniel Boland, and Stephen Brewster. Transient and transitional states: Pressure as an auxiliary input modality for bimanual interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 401–410, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2473-1. doi: 10.1145/2556288.2557260. URL `http://doi.acm.org/10.1145/2556288.2557260`.

Takashi Miyaki and Jun Rekimoto. GraspZoom: Zooming and scrolling control model for single-handed mobile interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, pages 11:1–11:4, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-281-8. doi: 10.1145/1613858.1613872. URL `http://doi.acm.org/10.1145/1613858.1613872`.

Kevin Nelson. Exploring Apple's 3D touch. `https://medium.com/@rknla/exploring-apple-s-3d-touch-f5980ef45af5`, 2015. Accessed: May 10, 2017.

Alexander Ng and Stephen A. Brewster. Investigating pressure input and haptic feedback for in-car touchscreens and touch surfaces. In *Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, Automotive'UI 16, pages 121–128, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4533-0. doi: 10.1145/3003715.3005420. URL `http://doi.acm.org/10.1145/3003715.3005420`.

Overcast Radio LLC. Overcast. `https://overcast.fm`, 2017. Accessed: July 13, 2017.

Mark E Payton, Scott J Richter, Kristopher L Giles, and Tom A Royer. Transformations of count data for tests of interaction in factorial and split-plot experiments. *Journal of economic entomology*, 99(3):1002–1006, 2006.

Sebastien Pelurson and Laurence Nigay. Bimanual input for multiscale navigation with pressure and touch gestures. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI 2016, pages 145–152, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4556-9. doi: 10.1145/2993148.2993152. URL http://doi.acm.org/10.1145/2993148.2993152.

Gonzalo Ramos and Ravin Balakrishnan. Zliding: Fluid zooming and sliding for high precision parameter manipulation. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, UIST '05, pages 143–152, New York, NY, USA, 2005. ACM. ISBN 1-59593-271-2. doi: 10.1145/1095034.1095059. URL http://doi.acm.org/10.1145/1095034.1095059.

Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. Pressure widgets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 487–494, New York, NY, USA, 2004. ACM. ISBN 1-58113-702-8. doi: 10.1145/985692.985754. URL http://doi.acm.org/10.1145/985692.985754.

Gonzalo Ramos, Andy Cockburn, Ravin Balakrishnan, and Michel Beaudouin-Lafon. Pointing lenses: Facilitating stylus input through visual-and motor-space magnification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 757–766, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9. doi: 10.1145/1240624.1240741. URL http://doi.acm.org/10.1145/1240624.1240741.

Gonzalo A. Ramos and Ravin Balakrishnan. Pressure-Marks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1375–1384, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9. doi: 10.1145/1240624.1240834. URL http://doi.acm.org/10.1145/1240624.1240834.

Jun Rekimoto and Carsten Schwesig. PreSenseII: Bi-directional touch and pressure sensing interactions with tactile feedback. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '06, pages 1253–1258, New York, NY, USA, 2006. ACM. ISBN 1-59593-298-4. doi: 10.1145/1125451.1125685. URL http://doi.acm.org/10.1145/1125451.1125685.

Christian Rendl, Patrick Greindl, Kathrin Probst, Martin Behrens, and Michael Haller. Presstures: Exploring pressure-sensitive multi-touch gestures on trackpads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 431–434, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2473-1. doi: 10.1145/2556288.2557146. URL `http://doi.acm.org/10.1145/2556288.2557146`.

Anne Roudaut, Eric Lecolinet, and Yves Guiard. MicroRolls: Expanding touch-screen input vocabulary by distinguishing rolls vs. slides of the thumb. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 927–936, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-246-7. doi: 10.1145/1518701.1518843. URL `http://doi.acm.org/10.1145/1518701.1518843`.

Robert J Schwarz. The anatomy and mechanics of the human hand. *Artificial limbs*, 22, 1955.

Kang Shi, Pourang Irani, Sean Gustafson, and Sriram Subramanian. PressureFish: A method to improve control of discrete pressure-based input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1295–1298, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-011-1. doi: 10.1145/1357054.1357256. URL `http://doi.acm.org/10.1145/1357054.1357256`.

Daniel Spelmezan, Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga. Side pressure for bidirectional navigation on small devices. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '13, pages 11–20, New York, NY, USA, 2013a. ACM. ISBN 978-1-4503-2273-7. doi: 10.1145/2493190.2493199. URL `http://doi.acm.org/10.1145/2493190.2493199`.

Daniel Spelmezan, Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga. Controlling widgets with one power-up button. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 71–74, New York, NY, USA, 2013b. ACM. ISBN 978-1-4503-2268-3. doi: 10.1145/2501988.2502025. URL `http://doi.acm.org/10.1145/2501988.2502025`.

Craig Stewart, Michael Rohs, Sven Kratz, and Georg Essl. Characteristics of pressure-based input for mobile devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 801–810, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-929-9. doi: 10.1145/1753326.1753444. URL `http://doi.acm.org/10.1145/1753326.1753444`.

Craig Stewart, Eve Hoggan, Laura Haverinen, Hugues Salamin, and Giulio Jacucci. An exploration of inadvertent variations in mobile pressure input. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '12, pages 35–38, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1105-2. doi: 10.1145/2371574.2371581. URL `http://doi.acm.org/10.1145/2371574.2371581`.

Faisal Taher, Jason Alexander, John Hardy, and Eduardo Velloso. An empirical characterization of touch-gesture input-force on mobile devices. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces*, ITS '14, pages 195–204, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2587-5. doi: 10.1145/2669485.2669515. URL `http://doi.acm.org/10.1145/2669485.2669515`.

Matthieu B. Trudeau, Tawan Udtamadilok, Amy K. Karlson, and Jack T. Dennerlein. Thumb motor performance varies by movement orientation, direction, and device size during single-handed mobile phone use. *Human Factors*, 54(1):52–59, 2012a. doi: 10.1177/0018720811423660. URL `http://dx.doi.org/10.1177/0018720811423660`. PMID: 22409102.

Matthieu B. Trudeau, Justin G. Young, Devin L. Jindrich, and Jack T. Dennerlein. Thumb motor performance varies with thumb and wrist posture during single-handed mobile phone use. *Journal of Biomechanics*, 45(14):2349 – 2354, 2012b. ISSN 0021-9290. doi: http://dx.doi.org/10.1016/j.jbiomech.2012.07.012. URL `http://www.sciencedirect.com/science/article/pii/S002192901200406X`.

Matthieu B Trudeau, Deanna S Asakawa, Devin L Jindrich, and Jack T Dennerlein. Two-handed grip on a mobile phone affords greater thumb motor performance, decreased variability, and a more extended thumb posture than a one-handed grip. *Applied ergonomics*, 52:24–28, 2016.

Daniel Vogel and Patrick Baudisch. Shift: A technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 657–666, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9. doi: 10.1145/1240624.1240727. URL `http://doi.acm.org/10.1145/1240624.1240727`.

Graham Wilson, Craig Stewart, and Stephen A. Brewster. Pressure-based menu selection for mobile devices. In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '10, pages 181–190, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-835-3. doi: 10.1145/1851600.1851631. URL `http://doi.acm.org/10.1145/1851600.1851631`.

Graham Wilson, Stephen A. Brewster, Martin Halvey, Andrew Crossan, and Craig Stewart. The effects of walking, feedback and control method on pressure-based interaction. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '11, pages 147–156, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0541-9. doi: 10.1145/2037373.2037397. URL `http://doi.acm.org/10.1145/2037373.2037397`.

Jacob O. Wobbrock. Practical Statistics for HCI. `http://depts.washington.edu/madlab/proj/ps4hci/`, 2017. Accessed: September 7, 2017.

Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 143–146, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1978963. URL `http://doi.acm.org/10.1145/1978942.1978963`.

Jinghong Xiong and Satoshi Muraki. An ergonomics study of thumb movements on smartphone touch screen. *Ergonomics*, 57(6):943–955, 2014. doi: 10.1080/00140139.2014.904007. URL http://dx.doi.org/10.1080/00140139.2014.904007. PMID: 24707989.

Index

Typeset September 28, 2017