Chair for Computer Science 10 (Media Computing and Human-Computer Interaction)

RWTH AACHEN UNIVERSITY

PullBand: An Elastic Force-Sensitive Wristband for Smartwatch Interaction

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

by
Tarek Hassan

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr.-Ing. Stefan Kowalewski

Registration date: 04.07.2018
Submission date: 13.12.2018

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

The smartwatch outperforms the smartphone in quick interaction scenarios. By using the smartwatch, the user can perform quicker interactions with a rise of a hand without the need to take his smartphone out of his pocket. However, the watch screen, being the main interaction possibility, suffers from various interaction problems. Its small sizes forces the fat finger and the visual occlusion problems. It also fails to support quick interactions while on the go as it requires visual attention from the user who has to stay still in order to accurately select the on-screen targets.

Most of the related work, which has extended the interaction of the smartwatch, has focused on the fat finger and the visual occlusion problems. Only a few papers have considered the usage context of the smartwatch for eyes-free and in-motion interaction. In this manner, related work has pointed out the importance of utilizing the band, taking advantage of the act of deformation, to offer an input modality for in-motion and eyes-free interaction. The main objective of this thesis is to explore, implement, and evaluate a deformational interaction technique on the watch band that tackles the interaction problems and the usage context of the smartwatch. In this work, we explore different deformational interactions on the band of the watch. We conduct a preliminary user study to find the most preferred technique and its characteristics on the band. In addition, we follow that by implanting the chosen prototype, PullBand, to support target selection tasks while in-motion and eyes-free. We evaluate our high fidelity prototype against and auditory menu while both in stationary mode and in-motion. Results may provide evidence that users could select audio targets with average success rates of 96% using three menu items and 94% using five menu items, while being in different motion levels. In addition, findings allow us to suggest guidelines that maximize the user performance in terms of *Time* and *Accuracy*, using PullBand.

Acknowledgements

First, I would like to thank all of the participants, who took part in the user studies, for their time and effort.

Secondly, I would like to thank my supervisor Christian Corsten, for his valuable support and insightful suggestions during the course of this thesis.

I would also like to thank my thesis advisor Prof. Dr. Jan Borchers, who provided us with the state-of-art environment, which highly motivated in me the passion to improve and learn.

A very special thanks to all the PhD, thesis, and HiWi students at the Media Computing Group. Being surrounded with smart motivated people who share a similar professional passion was an enjoyable experience, which I appreciate and will be always looking to have in the future.

Finally, I want to thank my mother, father, and my two loving sisters. You have always provided me with all the support and love that a person could ever ask for.

Tarek.

Chapter 1

Introduction

Over the past few years, the smartwatch has become increasingly popular providing an added benefit over the smartphone in performing quick and easily accessible interactions. By using the smartwatch, users can easily check notifications, play music, start a workout, or answer a phone call, without the need to take their smartphones out of their pockets.

Smartwatches simplify quick interactions.

Figure 1.1: The smartwatch provides micro interactions through a discrete set of quickly accessible commands.

The watch screen suffers from the visual occlusion and the fat finger problems.

However, the smartwatch offers to perform these interactions mainly through the touch screen which suffers from the fat finger and the visual occlusion problems [Xiao et al., 2014, Bi et al., 2013]. Theses problems made it difficult to have multiple target on the screen as users struggle to select them due to their small size and the fact that the finger covers over half of the screen, making it difficult to see which target is actually being selected.

The touch screen is not able to support in-motion and eyes-free scenarios.

More importantly, although the main benefit of the smartwatch is providing quick and easily accessible interactions, it still suffers to act when the user is in-motion or on the go. For example, when the user is walking in a street full of people and obstacles, he might want to play or stop a song, answer a phone call or decline it. Using the touch screen to perform these interactions, while in-motion, is not only difficult but could also cause safety issues [Mustonen et al., 2004, Schildbach and Rukzio, 2010]. Therefore, the user has to stop, to look at the screen, be still and then select the target accurately and safely [Marshall and Tennent, 2013]. Offering the possibility to perform quick interactions while in-motion and eyes-free would provide not only a much safer interaction but also an interesting experience, minimizing the interruption caused by the need to stop-to-interact [Marshall and Tennent, 2013].

Only a few papers have suggested techniques for in-motion or eyes-free interaction.

In general, most of the related work that have extended the interaction of the smartwatch has been only concerned with the fat finger problem and the visual occlusion problem [Baudisch and Chu, Oney et al., Lyons et al., Xia et al.]. Only a few papers have suggested supporting in-motion or eyes-free interaction scenarios. Out of these papers, Cheung et al. [2017] and Vogl et al. [2017] pointed out to the importance of utilizing the act of deformation on the watch band as a promising input modality in such interaction context. Taking advantage of the act of deformation not only provides a tactile feedback that is important for eyes-free interaction but also requires less accuracy from the user while in-motion. However, both of these papers have not evaluated their techniques in such interaction scenarios.

In this master thesis, we explore using the affordance of the watch band to create a novel interaction technique that not only tackles the visual occlusion and the fat finger problems but also facilitates in-motion and eyes-free interactions through the tactile feedback that it provides.

Consequently, this thesis is structured as follows:

The smartwatch is mainly used for micro interactions.

- In Chapter 2, we provide a brief look into the evolution of the smartwatch and the change of its usage context over the years. The chapter concludes by indicating the usage context of the current smartwatch and pointing out to the importance of supporting quick and micro interactions.

The user cannot perform micro interactions on the go, using the touch screen.

- In Chapter 3, we present how the main input modality for the smartwatch is represented by the touch screen. Depending on the touch screen, however, suffers from various interaction problems including not being able to support quick and micro interactions especially on the go. We also indicate the limitations of the other input possibilities that are available on the current smartwatches including the physical buttons, crown, bezel, and voice input.

We explore how related work has extended the interaction of the smartwatch.

- In Chapter 4, we indicate what researchers have done to extend the interaction of the smartwatch beyond the screen to one of four main areas: in air interaction, watch face, skin, and the wristband. We list some of the most important research projects and show how most of the related work has been only concerned with solving the visual occlusion and the fat finger problems, giving almost no importance to evaluate the input techniques in common usage scenarios such as in-motion and eyes-free interactions. We finish the chapter by focusing on previous work which lights up on the promising possibility of utilizing the wristband through the act of deformation to support in-motion and eyes-free interaction. This leads us to ask our first research question: *How convenient are different deformational interaction techniques on different locations on the watch band for in-motion and eyes-free interaction?*
- Chapter 5 starts by answering the first research question, exploring new possibilities of interactions that expand the input expressiveness of the smartwatch

We explore various deformational interaction possibilities.

to the wristband. We present the iterative design process of our low fidelity prototypes which support different kinds of deformational acts including bending, twisting, and pulling the wristband. Finally, the chapter concludes with the implementation details of four low fidelity prototypes.

Finding the most preferred interaction technique leads us to ask our second research question.

- In Chapter 6, we finish answering the first research question by evaluating our four low fidelity prototypes in a preliminary user study. Our objective is to find the user's preference not only for our deformational interaction techniques but also for the interaction locations on the wristband. This chapter refers to the study design, including hypotheses, and the study's task. In addition, results and implications are discussed. Finally, the chapter concludes with the most preferred location and prototype. Having identified the most preferred design leads us to ask the second research question: *How well could the user performs using this technique for in-motion and eyes-free conditions?*

We implement a high fidelity prototype and evaluate it against target selection tasks.

Chapter 7 starts answering the second research question by sufficiently completing the development of our most promising prototype to make it capable to perform menu selection tasks. Chapter 8 continues answering the question by evaluating the appropriateness of the final design in terms of user's performance and satisfaction. During this evaluation, participants are asked to select predefined targets as quickly and accurately as possible eyes-free and while in-motion.

- In the final chapter, we start by briefly summarizing the work and indicating the learned lessons. The chapter concludes with some limitations alongside suggestions for possible future work.

Chapter 2

The Smartwatch Evolution

In the following chapter, we take a brief look into the evolution of the smartwatch over the years; we point out the important changes that have occurred in its usage context since it was an idea and until today. We proceed to show how considering the usage context in the designing and the advertising process have played a significant role in the increase of adoption by the population. In this manner, we draw attention to the importance of micro interactions in the current smartwatches.

2.1 The Smartwatch Over the Years

The classical wristwatch has been one of the most widespread wearables of all time and seeing people interacting with their smartwatches has been socially accepted for almost 50 years [Buxton]. The next section provides a brief look into the evolution of the smartwatch since it was an idea in a comic book until the latest Apple Watch Series 4 [Apple, 2018].

The watch is the one of the most widespread wearables of all time.

2.1.1 Idea in a Comic Book

Dick Tracy carried a smart wrist communicator with video capabilities.

The world’s obsession with the smartwatch started in the 1930s when the famous comic book series ”Dick Tracy” was released [Roberts, 1993]. Tracy was a police detective, suited up in a yellow coat and known for the fact that he carried futuristic gadgets which helped him in his investigations. In one of these comic books, Tracy started to carry a smart wrist communicator with video capabilities. This was the first time that such a futuristic idea was introduced to the public. Tracy’s ”smart” wristwatch had a big popularity among the fans who started to ask themselves whether this idea was actually feasible; the answer at that time was simply ”No”. In fact, there was not any kind of wristwatch which had any functionality other than showing the time.

2.1.2 Early Smartwatches

In 1972, a brand of the American Hamilton Watch Company called Pulsar created the first digital watch which included a LED screen, making it possible for the first time to show digital numbers [Hochet and Acosta, 2002]. Although this new watch did not have any added functionalities other than checking for time, having a digital screen instead of the analog one, opened a new space for a design that has never existed before. Later in 1975, Pulsar added a calculator into the digital watch bringing for the first time a new functionality other than checking for time. Seiko TV Watch was launched in 1982, the first watch to incorporate a television [EPSON, 1982]. However, the watch needed from the user to plug his watch to a TV/radio receiver in the size of a Walkman to be able to watch the TV. In the following years, the brand Casio created many models which had an impact on the industry [Buxton]. Casio created the first Databank calculator watch CD-40 which did not only perform calculator functions as its predecessors, but also stored appointments, names, addresses, and phone numbers. Casio followed that with an analog model called AT-550-7 which supported touch input. The watch supported simple gestures as numbers and operators. Ca-

Casio Databank CD-40

Figure 2.1: Timeline: The development of watch usage over the years.

sio TC-50 came after that, offering a virtual keypad which allowed users to store addresses and calender entries on the watch.

In 2000, the IBM Linux WatchPad 1.5 provided a design which included a touch screen and a crown as a proof of concept [Narayanaswami]. It is worth to note that this design of the crown is similar to the currently available crown on Apple Watch. In 2003 Microsoft started to make what was called the ”SPOT watches”, making it possible for the watch to receive 1-way FM broadcast; the broadcast included localized weather service, news, and traffic information [Buxton].

In 2000, Linux Watch IBM WatchPad 1.5 provided a touch screen.

2.1.3 Smartwatch with Smartphone Capabilities

For the first time, at the Consumers Electronics Show (CES) 2009, the smartwatch which had been seen in the Dick

LG GD-910 with video calling capabilities.

Tracy comic book came to live when LG Electronics introduced the GD-910, the first 3G watch with video calling capabilities [Buxton]. However, this watch also came with many limitations in terms of interaction experience, screen resolution, storage size, and most importantly a high price tag leading to a relatively low adoption by the population. For the following few years, many firms introduced new watch models with novel and modified features. But they were still categorized as an extension for the smartphone. In the mean time, many fitness trackers started to sell on the market as they were offering a new way to monitor fitness, health, and lifestyle.

Smartwatches seemed not to offer an added benefit over smartphones.

In 2014, Google announced Android Wear, a version of Google's Android operating system designed for smartwatches, and started to ship it with various watch brands such as Motorola and Samsung. This offered developers a framework to develop new applications bringing smartwatches one step closer to smartphones. However, these smartwatches were still not able to bring a totally new functionality which the smartphone did not support. It seemed that the smartwatch would not have a clear long-term added benefit over the smartphone. Later and in 2015, Apple introduced the first Apple Watch which turned out to be the defining product in the smartwatch category for following years. It was neither a fitness band, a watch, nor a fashion accessory, despite taking a little bit from each. However, the first Apple watch still needed having an iPhone around, did not have a GPS tracking capability, and lacked a cellular connectivity.

The Workout App on the Apple Watch.

In the following years smartwatches started to become more independent from the smartphone and started to focus on supporting fitness and health tracking as they received features such as GPS tracking capabilities and waterproof support. These improvements started to give the smartwatch what it had always missed, an added value to the smartphone. Having fitness tracking moved the watch away from just being a way to get smartphone notifications on the wrist. With the later versions of the Apple watch, for example, the user could quickly start a workout session, check for his steps count, take calls, switch tracks without the need to take his phone out of his pocket.

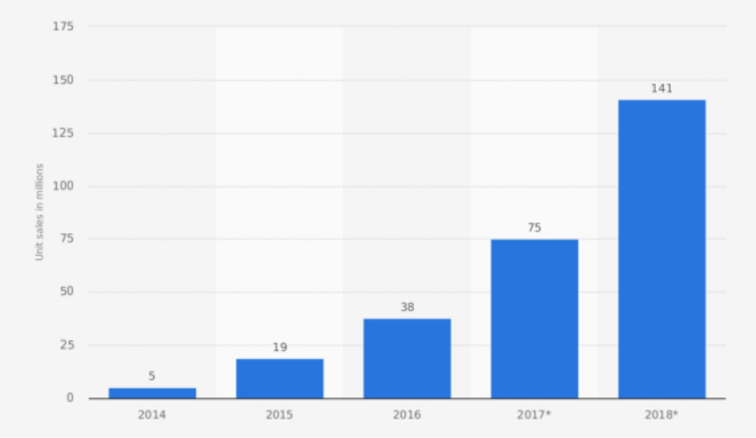


Figure 2.2: Estimation of the smartwatch unit sales worldwide from 2014 to 2018 (in millions) indicating exponential growth [Statista, 2017].

In general, smartwatch growth has increased dramatically leading to a market full of different smartwatch brands as Apple, Samsung, Fitbit, Huawei, Fossil Group, Garmin, and LG. Figure 2.2 shows an estimation of the increase of smartwatch unit sales between 2014 and 2018 with sales forecast to reach 141 million units in 2018 [Statista, 2017].

A dramatic increase in the smartwatch adoption in recent years.

It can be said that there is a correlation between the adoption of the smartwatch and the switch that the smartwatch made to support quick interactions, especially the finesse tracking capabilities, which the smartphone suffers to support. Still, researchers have been interested in analyzing how people use their smartwatches to identify new areas where the smartwatch is able to outperform the smartphone. Next section briefly discuss some of this work.

A correlation between supporting fitness tracking and the adoption of the smartwatch.

2.2 Micro Interactions: The Added Benefit Over the Smartphone

Several studies have been carried out on the usage of the smartwatch investigating the preferred applications and the difference in usage from the smartphone. Min et al.

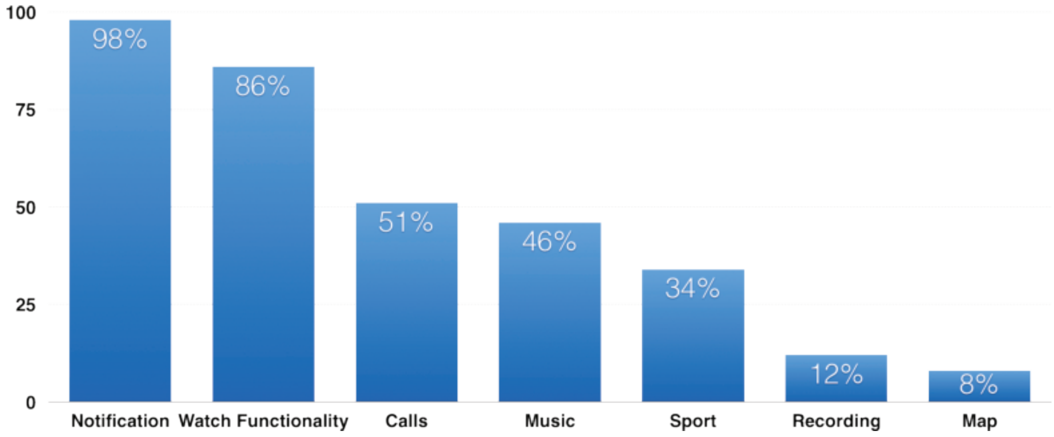


Figure 2.3: Type of applications users prefer to have on their wrist, notifications came in first place Min et al. [2015].

Smartwatch usage overcomes the disruptive nature of the smartphone.

[2015] asked users about the type of applications they prefer to have on their wrist. As seen in Figure 2.3, notifications came in first place followed, by watch functionality, calls, and music. Sport came only in the fifth place, which is understandable if we take into consideration that this study came in 2015 just right before smartwatches started to be commercialized for their fitness tracking capabilities as seen in the previous section. In general, the smartphone has been for long criticized for its disruptive nature in social context especially short interactions. In this manner, smartwatches overcome this issue by providing discrete list of micro interactions that can be performed quickly on the go.

Visuri et al. [2017] showed that 65.2% of smartwatch sessions are less than 5 seconds.

Another study by Visuri et al. [2017] focused on usage duration of the smartwatch and how it differs from its counterpart on the smartphone. Results showed that 65.2% of the sessions on the smartwatch are less than 5 seconds. These sessions included short interactions such as playing the next song or checking for arriving messages. Figure 2.4 shows that most smartwatch sessions, whether initiated by notifications or by the user, were much shorter than smartphone sessions. On the other hand, people can easily spend minutes and even hours on their smartphones accessing social media, browsing the Internet, watching videos, or reading articles [Visuri et al., 2017]. However, such scenarios are hardly possible for long durations on the smartwatch mainly due to the small size factor of the watch screen.

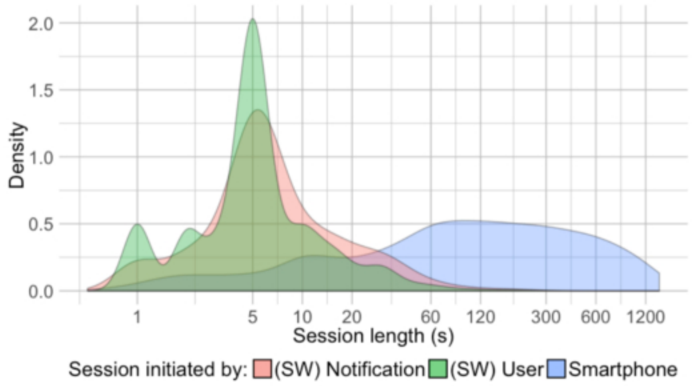


Figure 2.4: Usage Session length, (65.2%) of the sessions on the smartwatch are less than 5 seconds [Visuri et al., 2017].

Looking back at the results by Visuri et al. [2017] and Min et al. [2015], it is clearly obvious that there is a significant difference between the smartphones and the smartwatches in terms of usage duration and context. Accordingly, the development of the smartwatch has to focus on filling the areas where smartphones did not fit perfectly. In this manner, smartwatches development started to aim at focusing on short interaction scenarios.

It is important for the smartwatch development to focus on supporting micro interactions.

To summarize, this chapter began by describing the evolution of the smartwatch, highlighting the change of its usage context over the years. We have seen how current smartwatches support movement by reducing the frequency that a user has to get their smartphone out of their pocket to check a notification or change a song. Current smartwatches also support fitness and health tracking for activities which require a broad range of movements. It should be emphasized that these micro interaction could always happen anytime and anywhere regardless the user state weather he was standing or in-motion. However, the smartwatch interaction suffers to support these interaction while on the go. The next chapter discusses the limitation of the smartwatch interaction in general and especially focuses on its flaws in supporting in-motion interaction.

Chapter 3

The Input Modalities and Their Problems

In the last chapter, we presented the usage context of the current smartwatches in performing quick interactions. In this chapter, we start by exploring the touch screen, being the main input possibility for the smartwatch. We refer to its most common interaction problems and highlight its drawbacks in supporting quick interactions on the go. We conclude the chapter by briefly summarizing the alternative input possibilities on the smartwatch referring to their advantages and disadvantages.

3.1 The Touch Screen

The touch screen suffers from many problems.

The touch screen has been established to be the main possibility for interaction with the most up-to-date smartwatches. The touch screen has many advantages such as providing a rich graphical user interface and both input and output at the same time [Darbar et al., 2016]. However, the small size of the watch screen results in difficulties in the interaction, namely, the fat finger problem and the visual occlusion problem. The touch screen also suffer from the Stop-To-Interact problem [Marshall and Tennent, 2013]. These problems are discussed in the following section.

3.1.1 The Visual Occlusion Problem and the Fat Finger Problem

60% Visual Occlusion.

The visual occlusion problem occurs when the user places his finger on the screen causing a large part to be occluded. Xia et al. [2015] reported that the user's finger can easily occlude over half of the screen's size during an interaction. Likewise, the fat finger problem is related to the fact that the smartwatch icons are too small compared to the size of the user's finger, making pointing and selection tasks cumbersome and difficult in many cases [Bi et al., 2013]. Bi et al. ran a two-dimensional finger touch experiment and found that error rates are significantly high ranging from 25% to 66% with target's width from 4.8 mm to 2.4 mm.

The problems of the fat finger and the visual occlusion are assumed to exist together on the smartwatch.

The two problems we just mentioned come as two separate problems on the smartphone, as it has a much larger size factor than the smartwatch. In other words, one of these problems could exist without the other on smartphones. On the other hand, it has commonly been assumed that, in the smartwatch, the two problems are one assumption due to the small size factor, and subsequently would have to exist together. Accordingly, many available commercial smartwatch systems deal with these problems by designing larger widgets which facilitates the interaction process. However, a major problem with this methodology is that it leads to limited number of items which could be placed on the watch face at a time. This, in addition, forces a sequence of swipes and taps to move between different faces resulting in a slower interaction.

Extensive research has been carried out to improve the watch screen interaction.

Researchers have exploited different possibilities to improve and enhance the interaction on-screen. Examples include [Lyons et al., 2012, Oney et al., 2013, Xia et al., 2015]. Lyons et al. [2012] implemented a smartwatch prototype with multiple tiny touch screens to extend the available interaction surface. Oney et al. [2013] developed Zoomboard to improve pointing on the keyboard of the watch. Lyons proposed to improve text entry by applying iterative zooming to enlarge small targets such as the letters of the keyboard. Xia et al. [2015] presented NanoStylus, an approach which used a finger-mounted tip stylus to interact with the smartwatch. In his paper, he reported that NanoStylus supports high precision by enabling fast and accurate pointing on a smartwatch with almost no occlusion. Using this input possibility Xia et al. [2015] reported a drop down of the visual occlusion from 60% to only 16%.

NanoStylus Xia et al. [2015]

Having discussed the problems with the touch screen and how the research was carried out to improve the screen interaction, the next section highlights the substantial problem with the touch screen interaction itself regarding its unsuitability for in-motion interaction.

3.1.2 Stop-To-Interact Problem

Using touch screens, users need to stop, stay still before he interacts [Marshall and Tennent, 2013].

As seen in Section 2.2, smartwatch interaction is mostly useful in quick interaction scenarios where the user does not want to take his phone out of his pocket. These scenarios could easily happen during in-motion interaction. However, smartphone systems, in general, are described as ”stop to interact” systems because of their dependability on the touch screen. By this term Marshall and Tennent [2013] describe how most mobile systems require the user to stop, stay still, and look at his screen before he is able to interact. While these systems could be the best in many scenarios, it can not be denied that using them while moving is not only difficult but also could cause safety issues [Mustonen et al., 2004, Schildbach and Rukzio, 2010]. Marshall and Tennent [2013] noted that it is really difficult and sometimes even impossible to interact with the screen while the user is engaged in another activity. Simple in-motion activities such as walking could be dangerous while interacting with the screen and could result in high risks when not paying attention to the surroundings. The risks increase when performing more demanding activities such as running or cycling.

Therefore, screen interaction limits smartwatch applications that have relation to in-motion interaction as it collects data while the user is moving. For example, the workout application on Apple Watch allows the user to hit ”start” to calculate the distance of his walk or run. However, interac-

Interaction with smartwatch applications is designed to be done while the user is still.

tion with this kind of applications can only be done before and after the run itself using the graphical user interface of the smartwatch. The user has to start the workout application and then starts his actual activity, and once he finishes, he can finally turn off the application. We also think that the screen could even limit the user's experience of other applications that have no relation to in-motion interaction such as notifications and phone calls. Users can get notifications anytime and anywhere regardless of whether he is walking, sitting, or laying in his bed.

The cognitive load limits the user's ability to perform watch screen interaction.

Interacting with the screen while in-motion is particularly difficult as it depends on the user's cognitive load. The user has to perform two activities at once. First, he has to engage in an in-motion activity such as walking or running, and secondly, he has to perform an interaction on his screen. Engaging in in-motion activity places cognitive demand which differs between walking and running [Marshall et al., 2016]. Screen interaction places specifically high load as it requires the user sight for the interaction. The user could not be able to focus on the two different tasks at once even if he was physically able to perform them. For example, it is unsafe for the user to cross the street while scrolling on his screen [Marshall et al., 2016].

Marshall et al. [2016] argues that making it possible to interact while in-motion would have many advantages for the following reasons:

1. Being able to interact with the smartwatch while engaged in in-motion activities is an interesting experience in itself and gives beneficial results to the user, physically and mentally. People wear their smartwatches while walking, running, swimming, or working out. They can simultaneously change a song, check notifications, or check the number of kilometers that they have already walked.
2. It would minimize the interruption caused by the need to stop-to-interact, resulting in a more enjoyable experience.
3. It would provide a safer Interaction while in-motion.

Figure 3.1: Apple Watch Series 4 [Apple, 2018].

Having discussed the importance of providing a system which supports in-motion interaction, we will discuss in the following section the other alternative inputs which the current smartwatches use and analyze their advantages and disadvantages.

3.2 Alternative Input Possibilities

Traditional devices that supported in-motion provided physical tactile buttons.

Other alternative input possibilities available on the current smartwatches include the traditional physical buttons, the rotatable bezel or crown, and voice commands. Looking back at older small devices that supported two-way interaction while the user being mobile, we can notice a very common input possibility they used to share; physical buttons for specific functionalities. This could be found in devices such as the Walkman, Mp3 players, or the iPod Nano. In these devices, a specific button is used to play the next song and another to play the previous one. The physical buttons provide tactile feedback which makes it possible to interact while in-motion. However, placing too many buttons makes it confusing to reach and interact while in-motion. Subsequently, these systems work well eyes-free with only limited number of buttons for a limited number of commands. The current smartwatches have fewer buttons which have smaller size and which, most importantly, are no longer mapped to do only one thing as the old but-

Figure 3.2: Siri voice command to start an outdoor workout [Apple, 2018].

The current smartwatches have fewer buttons with editable functionalities.

tons on the older devices. However, these buttons have different functionalities depending on the app they are in. They are also discrete and can only detect whether the button is pressed or not. The crown and the bezel provide a continuous range of input but they are mostly used for on-screen interaction.

Voice commands support eyes-free interaction.

Using voice commands is another source of input for current smartwatches. Speech commands has been normally invoked by special keywords such as "Hey Siri" on the Apple Watch seen in Figure 3.2 or "OK Google" on Android wear [Google]. Apple Watch Apple [2018] Siri supports eyes-free voice commands after pressing on the side button of the watch. Such voice control facilitates checking schedule, notifications, emails and fitness functionalities without having to tap on the tiny screen. Similarly, the user can ask Android wear "what's on my schedule?" to see the next few entries from Google Calendar. Google Fit can also support many functionalities such as "track my run" to launch your smartwatch's tracking capabilities, or "what's my step count?" or "what's my heart rate?". However, it is very important to realize that these systems are poorly designed to use while moving outdoors in noisy environments. Environmental and wind noise would easily result in inaccurate recognition. Activities such as outdoor walking or running, which have background noise, make it hard for the system to interrupt the commands. Another issue with speech commands is related to privacy. Users are not always comfortable to say their actions out loudly. If

Voice commands function poorly outdoor resulting in inaccurate recognition.

the user is in a quiet environment where he is surrounded by friends or strangers he might not prefer to share his interaction with the people around him. The user could have private information or he could be afraid to annoy the people who surround him [Guo and Paek, 2016]. Last but not least, using voice commands in the current smartwatch systems requires an Internet connection which should be theoretically always available. However, in reality that is not always the case, especially in areas with little or no signal at all.

Speech recognition require Internet connection.

In summary, this chapter argued that the current input possibilities of interacting with the smartwatch do not take full advantage of the context of use of the smartwatch. Using the watch screen, user can interact only before or after the activity. It went on to suggest that there is a strong need for an alternative possibility for such interaction scenarios. The next chapter describes how researchers approached and extended the interaction of the smartwatch, and focuses on the proposed techniques and their ability to support in-motion and eyes-free interaction. In this manner, we will classify these methods according to the areas they have extended the interaction to. In addition, we will indicate whether they have considered eyes-free and in-motion interaction in their design and identify the most promising category to support in-motion and eyes-free interaction.

Next chapter, explores how researchers extended the interaction of the smartwatch.

Chapter 4

Related Work

In the previous chapter, we have presented the prominent input modalities for the current smartwatches including the touchscreen as the main possibility for interaction, along with other alternative input possibilities including the physical buttons, the crown, the bezel, and the voice input. We indicated the most important interaction problems related to the small size factor of the touchscreen and the lack of support for eyes-free and in-motion interaction. In this chapter, we consider how researchers have tackled watch screen limitations by extending the interaction to one of the following areas around the watch: in-air, watch face, skin, and wristband. Each of these areas provides an interesting design space to be explored. For example, in-air provides futuristic interaction possibilities: users can select a target with a quick in-air gesture of their finger, or they could tilt the wearing hand for one-handed interaction. On the other hand, the three other areas including the bezel, wristband, and the user's skin, all share the concept of ubiquitous computing, firstly mentioned by Wiser [1991], which is based on providing a hidden technology that is not noticed by the user anymore, i.e., transforming the passive bezel, skin, or wristband into an interactive controller for the smartwatch interaction.

Researchers have extended smartwatch interaction to In-air, watch face, skin, and wristband.

It is worth mentioning that most of the related work has been mainly concerned with solving the visual occlusion problem and the fat finger problem. Only a few papers

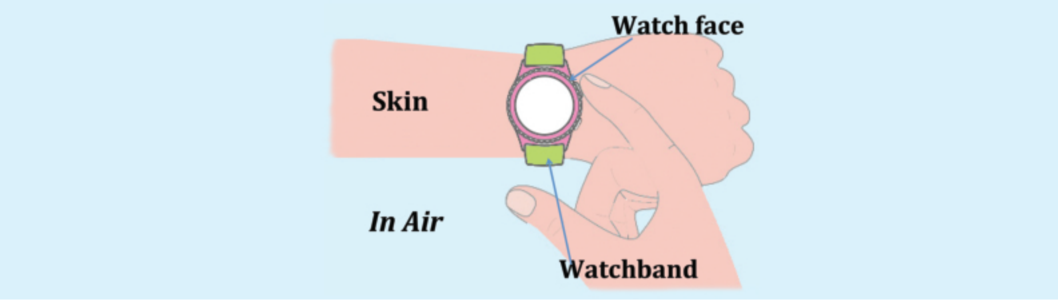


Figure 4.1: Interaction areas around the watch: in-air, the watch face, wristband, and the user's skin.

In Air	*Watch Face*	*Skin*	*Watchband*
🔴 Guo Objectpoint 🔴 Float 🔴 The Gesture Watch 🔴 HoverFlow 🔴 Abracadabra	🟢 Sidetap & Slingshot 🟢 Haptic wristwatch 🔴 PressTact 🔴 Xiao Mechanical Watch-face 🔴 WatchMI	🔴 SkinWatch 🔴 SkinTrack 🔴 SkinButtons	🔵 StretchEband 🟢 Exploring Eyes-free Interaction 🟢 Squeezy Bracelet 🟢 WatchIt 🟢 WRISTBAND.IO 🔴 NanoTouch 🔴 BandSense

🔴 No consideration for Eyes-Free/In-motion 🟢 Eyes-Free 🔵 In-motion

Table 4.1: The design space of related work. Most of the work that is related to support eyes-free and in-motion interaction falls into the watchband category.

Few papers considered in-motion and eyes-free interaction in their designs.

have suggested interaction techniques for in-motion and eyes-free scenarios (see Table 4.1). An even a smaller number have implemented working prototypes and only Perrault et al. [2013] has properly evaluated their technique in such eyes-free context. As we indicated in Chapter 3, although these problems are important to be solved, providing an input possibility which supports in-motion and eyes-free interaction is also an important aspect which should be taken into account when designing for the smartwatch.

In the following sections, we will go through each of the different interaction areas by giving examples of some of the important papers along with their main contributions and limitations. The chapter will also help to pinpoint open and a rich areas for further research still to be carried out.

4.1 In-air Interaction

In-air interaction is one of the most promising and futuristic techniques that extends the interaction of the smartwatch. It could be categorized into two key types: two-handed and one-handed interactions.

Two-Handed Interactions

In this approach, both user's hands are included in the interaction. The wearing hand of the smartwatch serves as an observing point to the other hand which could be moved around providing an input that is captured by special sensors inside the watch. Examples of such work could be found in the Gesture Watch by Kim et al. [2007] and HoverFlow by [Kratz and Rohs, 2009]. Both of previous examples have used proximity sensors to capture the movement around the watch. Harrison and Hudson [2009] used another technology to implement the same approach. In their work, titled Abracadabra, finger movements around the watch are captured using a magnetometer located inside the watch face. Using this concept, the user could scroll through an on-screen menu and click, in-air, to select the intended target. However, one limitation with this work is that the user always needs to wear a magnet on his finger.

Abracadabra, [Harrison and Hudson, 2009]

One important limitation of two-handed interaction is that it is unable to provide any tactile feedback. Consequently, the user might be less confident about the results of his actions especially eyes-free or while he is in-motion. Another major drawback of this approach is the limited number of gestures which could be recognized by the specified technologies [Kratz and Rohs, 2009].

Drawbacks include the lack of tactile feedback and a limited number of gestures.

Figure 4.2: Tilt-based interaction technique on the smartwatch [Guo and Paek, 2016].

One-Handed Interactions

This technique leaves one hand free for engaging in other life interaction scenarios.

One-handed (also referred to as hands-free) interaction is the second category that utilizes in-air interaction. In this approach, and unlike the two-handed one, the wearing hand serves not only as the activating hand but also as the observation point. This method leaves the other hand free and available to be used in other life interaction scenarios. Examples may include holding a cup of coffee or a suitcase while controlling the smartwatch. The most common one-handed interaction is tilt input. This technique was explored earlier on smartphones [Rahman et al., 2009]. Using a similar concept, Guo and Paek [2016] implemented a tilt-based interaction technique on the smartwatch where they directly mapped the position of a virtual pointer to the tilt angle of the smartwatch (Figure 4.2). Another example could be seen in the work of Sun et al. [2017] as they implemented a similar technique in their paper titled Float where they implemented tilting the smartwatch as a general and continuous 2D pointing method. In general, one-handed techniques depend on the smartwatch built-in sensors such as the magnetometer, the accelerometer, the built-in heart rate sensor, and the gyroscope.

Float, Sun et al. [2017]

The user has to keep his hand up and stable which is not suitable for in-motion interaction scenario.

It should be mentioned that both Guo and Paek [2016] and Sun et al. [2017] have evaluated their techniques using either on-screen target selection or navigation tasks along with providing visual feedback. None of these techniques have mentioned eyes-free or in-motion interaction scenarios. Actually, a significant drawback with one-handed in-

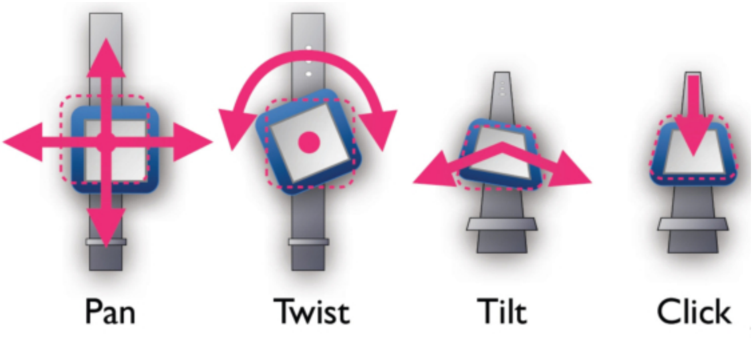
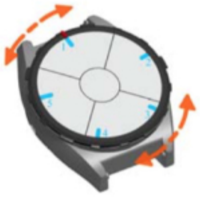


Figure 4.3: Multi degree of freedom mechanical watch face [Xiao et al., 2014]. The user can mechanically tilt, twist, rotate, or click the watch face.

teraction is that the user has to lift up his hand and keep it horizontally steady while he is performing the interaction. Therefore, the technique would arguably fail in-motion. Moreover, one-handed interaction, similar to the two-handed in air interactions, lacks any tactile feedback that is especially important in eyes-free interaction scenarios.

4.2 Utilizing the Watch Face



Haptic wristwatch utilized the bezel of the smartwatch.

There is a considerable amount of research which has focused on utilizing the watch face and the bezel around it. Pasquero et al. [2011] presented Haptic wristwatch, an early smartwatch prototype which utilized the bezel as a continuous input modularity for a gesture-based interface. Some of the interaction possibilities include covering the watch, turning the bezel, or swiping over the watch. However, Pasquero et al. [2011] focused primarily on the output rather than the input, i.e., they focused on the patterns of virtual haptic output that supports a simple eyes-free interaction.

Another interesting work that we found is by Xiao et al. [2014] who turned the standard static bezel to a multi-degree of freedom watch plate (Figure 4.3). They placed

Hall Effect sensors have a large size factor.

hall-effect sensors behind the watch face to provide mechanical input with three or more analog degrees of freedom. The user can mechanically tilt, twist, rotate, or click the watch face for on-screen interaction. The most notable drawback with this work is related to the large size factor of the hall effect joysticks [Xiao et al., 2014]. Another drawback could be the fact that including such mechanical setup requires redesigning the watch face.

Sidetap & Slingshot and WatchMI depend on the IMU sensors which fail in-motion.

Yeo et al. [2016a] presented Sidetap & Slingshot providing an eyes-free rapid navigation of a long list of items with a tapping or pressing on the edge and then releasing quickly. Yeo et al. [2016b] also presented WatchMI which sense continuous rate-based “touch pressure”, “twist angle”, and “pan movement” on the watch face. However, both of the previous techniques use the Inertial Measurement Unit (IMU) inside the watch, making it unsuitable for in-motion interaction.

PressTact focus on on-screen interaction.

Darbar et al. [2016] presented PressTact using side pressure-based input for Smartwatch Interaction. PressTact enables users to input different levels of pressure that can be used for bi-directional navigation (zooming, scrolling, and rotation) on smartwatches. However, Darbar et al. [2016] focused only on on-screen interaction.

4.3 Utilizing the Skin

Researches have also expanded the interaction of the smartwatches to the user's skin around the watch through various sensing techniques [Laput et al., 2014, Ogata and Imai, 2015, Zhang et al., 2016a,b].

SkinButtons is limited to the number of projected icons on the skin.

Laput et al. [2014] presented SkinButtons (Seen in Figure 4.4), a laser projected interface that enables input using projected light buttons and photo sensing techniques. They used infrared sensors to detect when the user selects the projected icons by pushing his finger on those icons. A noticeable disadvantage with this method is the limited number of commands that it provides which is limited to the number of icons that could be projected on the user's wrist.

Figure 4.4: Skin Buttons [Laput et al., 2014]. User selects the projected icons on the skin by pushing his finger on them.

TapSkin by Zhang et al. [2016a], made it possible to recognize up to 11 distinct tap gestures on the skin around the watch using the inertial sensors and the microphone of the smartwatch. However, this technique suffers in-motion as the inertial signals as well as the accuracy of the tapping locations start to be heavily affected. Zhang et al. [2016b] presented SkinTrack which enables continuous finger tracking on the user's wrist surface once the user's finger touches the skin. However, with such a technique the user has to consistently wear a special ring.

SkinTrack requires the user to wear a special finger in order to work.

4.4 Utilizing the Watch Band

Utilizing the watch band for input is another popular technique that has been explored in the context of smartwatch interaction. We divide the work in this area into two main categories. The first category deals with basic interaction techniques such as using touch gestures, applying pressure, or placing physical buttons on the band's surface. The second category takes advantage of the physical properties of the wristband, which supports the act of deformation, in order to provide tactile feedback.

We identify two interaction categories: basic or traditional, and deformational.

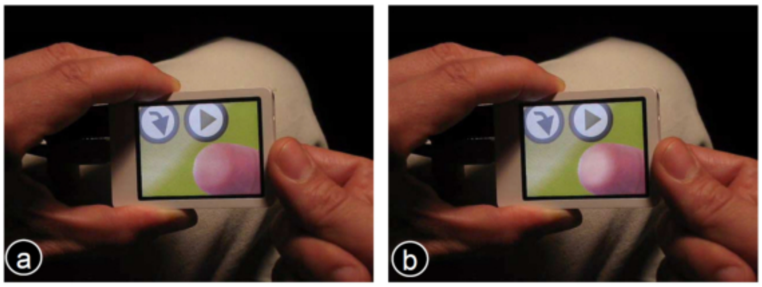


Figure 4.5: NanoTouch improves pointing accuracy on touch screens by using a touchpad on the back side of the device [Baudisch and Chu, 2009].

4.4.1 Basic or Traditional Techniques

In this category, we mainly focus on the work that has utilized the surface of the wristband to support basic interaction techniques such as using touch gestures, applying pressure, or placing physical buttons on the band's surface.

NanoTouch illustration on a watch band.

Baudisch and Chu [2009] suggested in their paper NanoTouch improving the pointing accuracy on the touch screen by using a touchpad on the back side of the device as seen in Figure 4.5. They also suggested placing a touch-pad on the back of the wristband near the clasp. In the same context, Saviot et al. [2017] implemented WRISTBAND.IO, a touch-pad on the back of the wristband. However, results demonstrated an increase in both user's frustration and task completion time compared to the stranded touchscreen interaction. In addition to using a touch-pad, Saviot et al. also utilized their prototype with haptic tangible buttons on the band for quick eyes-free input where each button can be reconfigured for frequently used functions. However, it is worth mentioning that they have not done any evaluation for this prototype.

WRISTBAND.IO

Another work could be seen by Perrault et al. [2013] who investigated using touch and sliding gestures on the surface of the wristband. Perrault et al. explored two novel interaction techniques for eyes-free interaction: the first technique implemented a button like input, and the second one demonstrated utilizing the band to act as a simple slider for

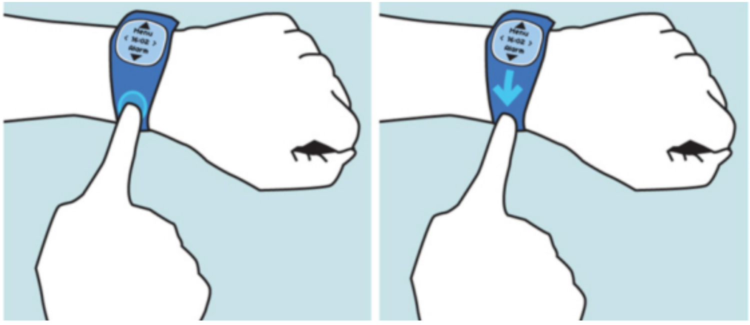


Figure 4.6: Watchit: user can perform touch and sliding gestures on the surface of the wristband for eyes-free interaction scenario [Perrault et al., 2013].

a continuous scrolling task. In addition, Perrault et al. evaluated these techniques for target selection tasks in eyes-free interaction scenario using audio feedback. In their prototype, the user can directly point to a specific zone on the band to hear an audio feedback. The user can select the intended target by raising his finger. If the user needed adjusting his selection, he can reach the right zone by sliding in the appropriate direction. Results of the user study confirmed the effectiveness of this technique in eyes-free usage scenarios.

Watchit is one of the few papers which evaluated their technique in eyes-free interaction scenario.

Another possibility of utilizing the surface of the wristband could be using pressure input. Ahn et al. [2015] explored using the pressure on the band surface to design a novel interaction technique. They suggested in their paper BandSense, a pressure sensitive multi-touch interaction on the wristband using custom pressure sensors. However, they did not implement any prototype nor conducted any evaluation.

Ahn et al. [2015] did not evaluate their pressure sensitive wristband.

Having discussed the first category of wristband's interaction using touch and pressure gestures, we move on to consider using the affordance of the band in designing novel interaction possibilities.

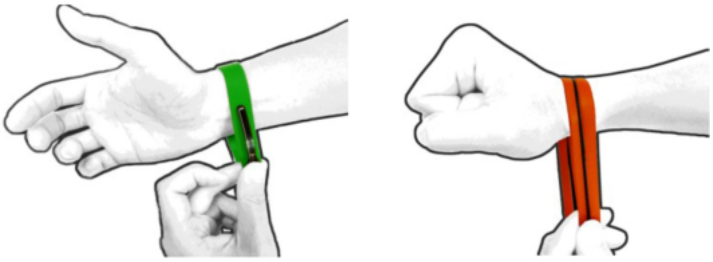


Figure 4.7: Cheung et al. [2017] contributed various sketches for deformable wristbands prototypes.

4.4.2 Deformational Interaction Techniques

For long, objects affordance have been used to create novel interactions.

In general, the term ”affordance” refers to the fact that physical objects have shape, material, and size properties which suggest to the user how he could interact with them [Dix, 2009]. In this manner, researchers have used the affordance of smartwatch bands to create novel interaction possibilities.

Flexible bands afford deformable interactions.

Stretch-sensitive wristbands Vogl et al. [2017]

Leather, stainless steel, and nylon are some of the most common materials that are used to create smartwatch bands. Each of these materials provides a different kind of affordance. Flexible bands, among others, have the ability to support the act of deformation. For example, leather wristbands provide an affordance to be bent or twisted whereas stretch weaved elastic bands have an affordance to be stretched and pulled. In this manner, Cheung et al. [2017] presented low-fidelity deformable wristbands prototypes and an interaction language applicable to these devices (Figure 4.7). Cheung et al. [2017] suggested using their prototypes for eyes-free interaction but they have not implemented or evaluated their proposed techniques. Using the act of deformation to create a novel interaction technique could be also seen in the work of Vogl et al. [2017] who presented, in StretchEBand, a new fabrication method to create stretch-sensitive wristbands. They suggested stretching for continuous input like list scrolling or sliding. They indicated that such a technique would offer ”imprecise gestures” which would suit micro interactions while users are in-motion. However, they only gave examples for controlling a stopwatch or music player while running but have not evaluated their prototype in-motion or eyes-free.

Open Space for Further Research

As we have seen in the previous section, researchers have explored various possibilities for interaction with the smartwatch. We have equally clarified how each of these methods has its strengths and its weaknesses. We believe that the best interaction experience is attained when each method is allowed to be used at its best, within an environment that allows for these techniques to merge without conflicts. Having said that, we strongly emphasize the importance of supporting in-motion and eyes-free interaction for the smartwatch, which many of these techniques lacked. Through our dive into the related work we have identified utilizing the act of deformation on the wristband as a promising input modality that supports eyes-free and in-motion interaction. Cheung et al. [2017] pointed out the reasons behind that:

The act of deformation has the ability for supporting in-motion and eyes-free interaction.

- **It provides tactile feedback through force and tension:** the act of deformation provides non-visual and tactile feedback through force and tension. When the user stretches an elastic band, a tension started to be felt not only on the skin of the wearing hand but also on the activating hand. Such type of feedback could be felt when we tie our shoelaces giving us the feeling that we successfully made the tie.
- **It provides two-handed feedback:** some techniques such as stretching or bending the watch band are two-handed as both hands are involved in the interaction. This allows the wearing hand to perform as a counter or balancing force to support successful deformation. It should be mentioned that not all deformation interactions require two hands. For instance, in Squeeze bracelet by Pakanen et al. [2014], the user needs only

one hand to squeeze the air capsule while the other hand can not influence the interaction at all.

Our technique aims to be not only eyes-free and in-motion specific but also capable of supporting screen interaction.

To sum up, this chapter has demonstrated the various areas that research considered in extending the interaction beyond the smartwatch's screen. We have briefly mentioned their advantages and disadvantages. We were primarily interested in the work related to the watch band and utilizing the act of deformation. We are determined to pursue our work in implementing a technique which utilizes the act of deformation on the watch band. This will allow us to design an interaction technique that does not only avoid the visual occlusion and the fat finger problems but also supports in-motion and eyes-free interaction. Having described the main objective of this thesis, the next chapter moves on to discuss the design process of our low fidelity prototypes.

Chapter 5

Exploring and Prototyping

In this chapter, we explore different deformable interactions possibilities that expand the input of the smartwatch to the watch band. We indicated in Chapter 4 how such kinds of deformable interaction possibilities not only avoid the problems of touchscreen interaction but also provide a tactile feedback that supports in-motion and eyes-free interaction.

We explore different techniques that support the act of deformation.

We start by exploring and testing some of the available sensors that support measuring different acts of deformation such as stretch and bend sensors. Understanding how the sensors work at an early stage helps us to come up with additional design ideas. Furthermore, the chapter discusses the design and the iteration process over low fidelity prototypes that support either stretching or bending and twisting. During our iteration process, we explore various layouts, locations, and physical affordances in order to finally decide on a specific set of prototypes that will be evaluated in a preliminary user study.

Before starting our design cycle, we briefly explore possible sensing techniques.

5.1 Exploring Sensors

This exploration aims to come up with new interaction ideas.

In this section, we examine different types of sensors which have the ability to measure either stretching or bending and twisting. It should be mentioned that our purpose, at this stage, is purely exploratory and is not intended to favor one technique over the others. In other words, we use this exploration to come up with new ideas that could offer a deformable interaction technique on the band of the smartwatch.

5.1.1 Stretch Sensors

Commercial Stretch Sensors

Images Scientific stretch sensor can tear apart when being stretched.

Looking at the available commercial sensors, we found two different brands that measure pulling or stretching. The first sensor is from ″Images Scientific ″ with product code ″RB-Ima-14″ is a thin elastic cord 1.5 mm in diameter. It measures stretching by measuring the change in its resistance when being stretched. The second brand of sensors is from ″Adafruit″ website with Product Id ″519″. These sensors rely on the same concept as the first ones but they are available in 1 meter in length and 2 mm in diameter.

It should be mentioned that our testing showed that the Images Scientific sensors could easily tear apart when being stretched making them unreliable to be used. On the other hand, the thicker diameter of the Adafruit sensors makes them more reliable to be stretched without tearing apart, but with a much smaller output magnitude.

Stitch-Based Elastic Sensors

Vogl et al. provides design guidelines for stitch-based elastic sensors.

Stitch-based elastic sensors are the second type of stretch sensors which we explored. Unlike the commercial based ones, theses sensors could be manufactured using home sewing machines. Vogl et al. [2017] provided a detailed explanation of how to design such sensors using conductive yarns along with the expected behavior.

Our test showed these sensors to change their behavior over time.

However, Vogl et al. did not provide any information on the reliability and the output range of these sensors. To investigate that, we created similar sensors using the design guidelines from Vogl et al. and tested how they work. The results showed a change in the sensor behavior over time. The reason for this change could go back to the stitched conductive lines which start getting loose after multiple stretches.

5.1.2 Bend Sensors

Flexpoint bend sensors detect bending and twisting.

Bend sensors consist of a coated substrate and are able to measure bending by measuring the change of their electrical conductivity once they are being bent. We were specifically interested in the sensors from Flexpoint Sensor Systems as they are provided in small sizes, one inch in length, suitable for the band. These sensors can also detect bidirectional bending which could be used to detect twisting [Shorey and Girouard, 2017]. The concept works by using a pair of these bend sensors in a crossly aligned manner. This alignment forces one of the two crossly aligned sensors to be bent in one direction as the other is being bent in the opposite direction [Shorey and Girouard, 2017]. Our testing showed that the sensors function as expected. However, repetitive bending and twisting caused the sensors to stop working. We doubt that the conductive circuits inside the sensors get damaged by multiple bending. Shorey and Girouard also shortly mentioned such unreliability in their user study.

Exploring the sensing techniques inspired us with additional design ideas.

Exploring how stretch and bend sensors work inspired us to create new interaction possibilities for the smartwatch. The stretch sensors from Adafruit design allow many design ideas that could take advantage of aligning different numbers of them on the band. They could also provide different grasping affordances depending on the way they are attached to the band. In addition, the small bend sensors could allow for the detection of bending and twisting on small areas of the band. The following sections of this chapter go through the design process of our low fidelity prototypes.

5.2 Iterating Over Low Fidelity Prototypes

We follow an iterative prototyping process.

In general, it is difficult to get the design right from the first attempt and it is necessary to have an iterative prototyping process to reach the final design. Dix [2009] pointed to the importance of prototyping in his book ”Human Computer Interaction”. He argues that: ”Iteration and prototyping are the universally accepted best practice approach for interaction design”. Therefore, we followed the following steps purposed by Dix:

1. **A good starting point**

   Exploring the related work and the sensing techniques helped us in our brainstorming sessions.

   We started our iterative design process by carrying out brainstorming sessions in which we explored and sketched various design ideas that could facilitate bending, twisting and pulling. We revisited the related work to find inspirational ideas. In addition, we used our experience with the sensors to come up with novel techniques that support the act of deformation.

2. **Understanding what is wrong with the design and trying to improve it in the next iteration**

   We collected user's feedback informally during our iterative design process.

   Throughout this chapter, we present how we evaluated our design ideas and only kept the most promising ones. We indicate the designed low fidelity prototypes and the feedback from informal user studies. User feedback helped us to get an outside prescriptive for our designs at an early stage.

During the course of this chapter, we first discuss the design process of the pulling prototyping and then move on to discuss the bending and twisting ones.

Figure 5.1: PullBand Vs. PullHandle.

5.2.1 Pulling Prototyping

Inspired by the thin elastic stretch sensors we explored before, we decided to use thin elastic cords in our exploration process for implementing pulling. Using this small thin factor helps us to explore pulling on any kind of wristband. For example, they could be attached to stretchable wristbands that already have an affordance for stretching and pulling. Equally important, they could be also attached to wristbands that do not have any affordance for pulling or stretching such as rubber, leather, and stainless steel bands. Moreover, having these thin elastic cords allows us to explore further possibilities of providing multiple cords on the band. It also allows us to explore different kinds of pulling affordance depending on the way we fix these cords on the band surface as we will see in the following sections. It should be mentioned that, for rapid prototyping purposes, we placed these thin elastic cords on Apple Nike smartwatch band which has small holes inside it, making it easy to attach the cords to.

We attach our elastic cords on Apple Nike band.

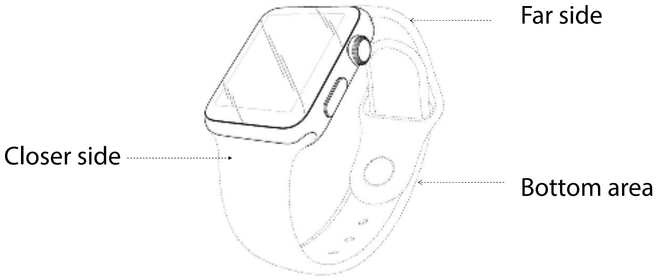


Figure 5.2: Interaction locations

Exploring Design Possibilities

In this section, we discuss the different design factors that we took into account during the first iteration of our physical prototyping process. In this manner, we refer to providing different pulling affordance, chord's count, and utilizing different locations on the band.

The way we attach the cords might affect the pulling direction preference.

1. **Pulling Affordance:** we explored two different pulling affordances depending on how we fixed or aligned the cords on the band. We wanted to investigate whether having these two different affordances would affect the degree of freedom of the pulling directions.

   In the first pulling affordance, the user slides a finger under the stretchable band and pulls it. The stretchable cord is attached to Nike watch band from its both ends as shown in Figure 5.1. In this way, the user is able to grasp and pull the cord at the middle part between the two fixed points. We will refer to this alignment using the term "PullBand". In the second alignment we explore having a free ended stretchable cord that could be attached to the band from one end. We

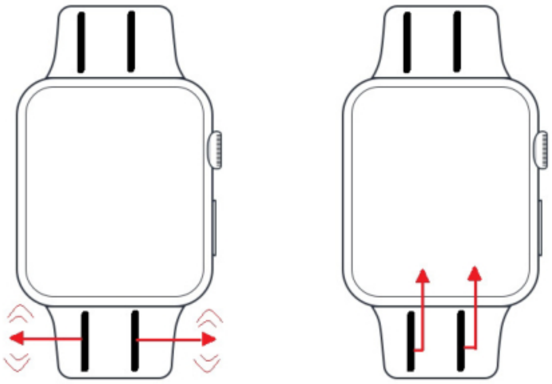


Figure 5.3: We explore different interaction possibilities with placing multiple parallel cords.

were interested in seeing whether having this alignment, which provides a free ended cord available for grasping, would give more freedom to users to pull the cord in multiple directions over the standard setup. To refer to this alignment we use the term ”PullHandle”. Please note that Figure 5.1 shows the final iteration of PullHandle (after adding magnets), as we will see in the final section of this chapter.

We explore three different interaction locations on the band.

2. **location:** in general, previous research by Perrault et al. [2013] referred to three areas on the band that could be easily perceived for interaction: closer side, bottom area, and far side. However, Perrault et al. did not explore these areas neither for deformable interactions nor while the user is in-motion. Therefore, it could be possible that the reachability of these locations is affected by the user’s hand posture which changes between sitting, walking, or running.

We explore placing multiple cords on the band.

3. **Count:** adding multiple parallel cords gives a larger set of commands. However, having multiple cords on the small surface of the band could also be annoying for interaction especially while in-motion or eyes-free. To find out users preferences regarding this issue, we designed prototypes with two cords as seen in Figure 5.3.

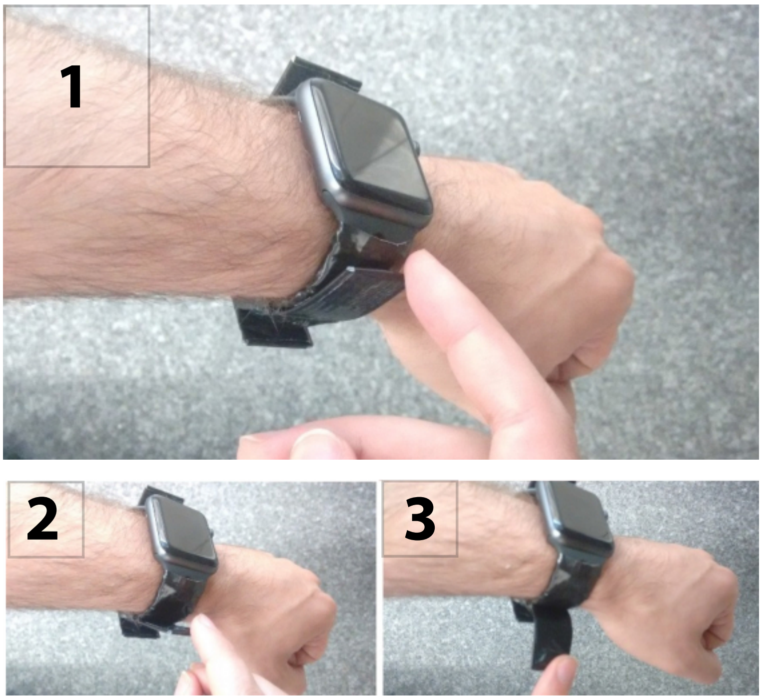


Figure 5.4: Bending a flexible strap towards the bottom.

5.2.2 Bending and Twisting Prototyping

Free elastic end

It is common for flexible traditional bands to have free popping elastic ends on the far side. We thought of using the affordance of this elastic end to support bending and twisting. However, after brief exploration we realized that the elastic end is not always available for interaction. It actually has a limited degree of freedom depending on the size of the user's wrist, making it sometimes unavailable for bending and twisting. In addition, the location of this free end depends highly on the wrist size of the user and could vary from the far side to the bottom area.

Inspired by this idea and to overcome the limitations that come with it, we thought of providing an extra dynamic surface that could be attached to the original band. This extra surface is always available to be bent or twisted. We used the surface of the original smartwatch's band as the base fixed surface and then we placed a flexible plastic part

Figure 5.5: Bending a flexible strap towards the top.

on it. We fixed these parts only with one of their ends leaving the other end free to be twisted or bent as seen in Figure 5.4, and Figure 5.5.

Exploring Design Possibilities

1. **Location:** we previously laid out the stretchable cords on three different areas of the band, namely, closer side, bottom area, and far side in order to explore the preference for each of these areas eyes-free and in-motion. We followed a similar procedure to also explore the use of these areas to facilitate bending/twisting eyes free and while in motion. Therefore, we placed a FlexibleStrap on each of these areas.

We place the flexible straps on three different locations on the band.

2. **Affordance:** the FlexibleStraps could be available for grasping on only one end as the other end is fixed to the band. Depending on which end is available for interaction the user could either bend the end toward the top or towards the bottom. We want to investigate whether this design would have any effect on the reachability and the simplicity of performing the interaction while in-motion.

The straps are fixed to the band at only one of their ends.

5.3 User Feedback and Enhancements

We collected informal user's feedback regarding our designs.

To evaluate the various factors that we presented, we designed several low fidelity prototypes that cover the selected factors and presented them to multiple users in an informal manner. The main purpose was to have an outsider look at our designs in order to define the most important factors that should be changed or formally tested in a user study. We made it clear to our participants that we wanted to discard undesired designs, variables, and ideas, only keeping the promising ones.

Feedback suggested eliminating further investigation of the number of placed cords.

Users' comments made us continue our exploration for location preference and different affordance that the prototype provides. On the other hand, we eliminate having multiple cords as users strongly expressed their dislike for having more than one. The user could easily grasp the wrong cord if he is not looking or in-motion. Furthermore, users pointed to some problems regarding the PullHandle cords thin endings, saying that the cords did not actually allow the affordance for pulling. In addition, these endings hanging freely from the band made the users uncomfortable. To fix these problems we decided to add small circular magnets to the free endings making them easy to grasp and forcing them to snap back to the band surface where thin magnets were also placed.

We added magnets to the end of the strings.

We also added magnets to the FlexibleStraps.

Moreover, for the bending and twisting prototypes, users reported that the FlexibleStraps stayed hanging after they have been bent, causing discomfort. To fix this problem we used a method similar to that used with the pulling prototypes; we added magnets on the free endings of the straps forcing them to snap back to their original position after bending.

As a result, we ended up with the following low-fidelity prototypes ready to be evaluated in a formal user study:

1. **PullHandle:** Band with three pull handles that the user can pull from the magnets at their ends. We place a PullHandle at each of the closer side, far side, and bottom area of the band.

2. **PullBand:** Band with one stretchable cord placed all around the original band circularly leaving a gap between the cord and the band for grasping.
3. **FlexibleStraps towards the top:** The user can bend the straps from their upper part towards the bottom.
4. **FlexibleStraps towards the bottom:** The user can bend the straps from their bottom part towards the top.

Chapter 6

Preliminary User Study

In this chapter, we evaluate the *user preference* regarding pulling, bending, and twisting.

In this user study, we evaluate our four low fidelity prototypes which we have introduced in Chapter 5. Our four prototypes simulate using pulling, bending, and twisting the watch band as possible ways of interaction with the smartwatch. Two of our prototypes facilitate pulling: PullHandle and PullBand, and the other two, FlexibleStraps, facilitate bending and twisting.

At this stage, we do not provide any visual, acoustic, or haptic feedback.

Please be reminded that this user study is part of the iterative design process which we follow to reach our high fidelity functioning prototype. Therefore, in this user study, the only feedback that our low fidelity prototypes provide is the tactile feedback resulting from the act of deformation. Any other visual, acoustic, or haptic feedback is not yet provided.

No significance test is performed in this preliminary user study.

Throughout this preliminary study, we capture how the users interact with our prototypes and analyze their satisfaction levels using a Likert scale questionnaire. In addition, the results of this user study are explained using descriptive statistics and no significant testing is performed. The results give us an idea about the most preferable technique (between bending, twisting, and pulling) along with the most preferable interaction location on the band in both in-motion and eyes-free usage context. The results of this study will be later used to implement our final prototype.

This chapter includes the hypotheses, the utilized task, the design, data collection, results, and implications. Independent as well as *dependent variables*, along with the experiment's target group, are stated. Finally, the chapter concludes with a statistical analysis of the results, and discusses the implications to reach the final prototype.

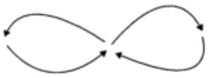
6.1 Hypotheses

Throughout the study, we examine the following hypotheses while the user is in-motion and not looking at the band (stated in null form, i.e., expected to be rejected):

H1 User preference is the same among different locations of the smartwatch's band.

H2 User preference is the same among different interaction techniques that support the act of deformation of the smartwatch band.

6.2 Task



Participants are asked to walk in figure eight.

We asked our participants to wear each of the low fidelity prototypes which we designed. Then we asked them to walk in what could be referred to as a figure eight 6m in length and 4m in width. A chair is placed in each of the holes of the figure 8 to represent an obstacle. Interacting with the band while walking in figure eight helps to simulate in-motion interaction scenarios as the user's cognitive load is split between the interaction and focusing on his path so he does not bump into one of the obstacles.

The study is divided into two main blocks, each divided into two sub-blocks.

We wanted to test prototypes which support pulling and others which support twisting and bending. Therefore, we divided the study into two main blocks where one block represents pulling and the other represents bending and twisting. Each of these block was further divided into two sub-blocks where each one represents one of our four proto-

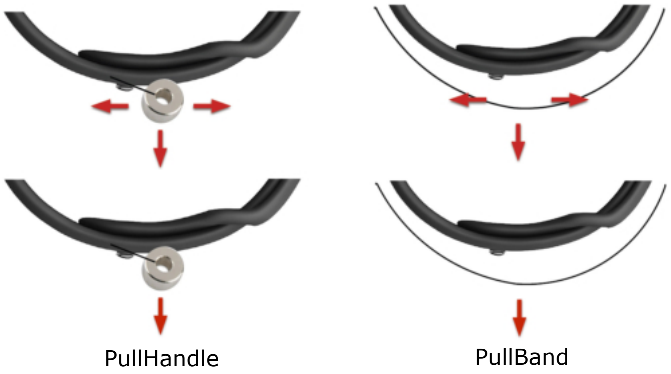


Figure 6.1: Pulling in multiple directions Vs. one direction.

types that are explained in Section 5.3. The order of both the main blocks and the sub-blocks was balanced using Latin square [Cochran and Cox, 1950]. Balancing was especially important in order to minimize the carry over effects that result from users' fatigue. After the participants finished all the four blocks, we asked them to answer a general questionnaire.

6.3 Design

6.3.1 Independent Variables

1. Pulling
    - Pulling affordance: (PullBand, PullHandle).
        (a) Direction: (multiple directions, one direction (orthogonal)). Please refer to Figure 6.1.
        (b) Location: (closer side, bottom area, and far side).
2. Bending and Twisting
    - Bending affordance: bendable straps (towards the top, towards the bottom).
        - Location: (closer side, bottom area, and far side).

6.3.2 Dependent Variables

We measure users' *Satisfaction* in terms of comfort, reachability, and usability in respect to the independent variables. In addition, we collect users comments regarding their *Technique Preference* and the interaction design characteristics they prefer.

6.3.3 Participants

Sixteen participants, all males, took part in the study.

We recruited sixteen users to participate in our preliminary user study. Participants were all students between 20 and 29 years old. We recruited only male participants since the bands which we used suited men wrist's size [170..195] mm whereas women wrist's size is usually smaller [140..170] mm, as mentioned by Apple Watch bands design guidelines[Apple, 2018]. We do not believe that having only males participants affected our results. However, we will add female participants in the final user study, in this thesis, when we evaluate the performance of the users with a high fidelity prototype.

6.4 Data Collection

We present a post-block questionnaire with a five point Likert scale.

After each block, we asked the participants to take a seat and to fill out a questionnaire (Appendix A). Having a post-block questionnaire helps our participants to answer questions more accurately than having a post study questionnaire especially considering that there were many tasks per block [Lazar et al., 2017]. In addition, sitting down and answering the questionnaire gives a chance to our participants to recover, so they do not get fatigued. We aimed with this questionnaire to collect users' *Satisfaction* for the techniques and the locations of the band using the specified technique. We used a five point Likert scale ranging from 1.0 “strongly disagree” to 5.0 “strongly agree”.

In addition, we asked participants to think aloud while they are performing the tasks in order for us to get a more comprehensive insight into their thoughts [Lewis, 1982]. The collected comments help us to find problems with the designs along with suggestions for improvements.

6.5 Study Results

We demonstrate the user *Satisfaction* levels using box plots. In general, box plots are able to offer insight into the distribution of users' responses and can easily convey the agreement or the disagreement over the specified condition. This could be easily interpreted from the graphs by distinguishing comparatively short box plots, which emphasize agreement between the users; and tall box plots which emphasize disagreement. In this analysis, we are mostly interested in the results where users hold similar opinions over a condition, i.e., short box plots.

Box plots helps to convey an agreement or a disagreement between users.

We introduce the results for each of the interaction techniques, one at a time. For each technique, we indicate user *Satisfaction* level regarding the respected independent variable for the specified technique.

6.5.1 Pulling

Figure 6.2 shows the results of users *Satisfaction* levels for pulling according to all the independent variables (direction, location, and affordance).

Pulling the PullBand at the bottom area was the most preferred.

Looking at the figure, we can easily indicate that the only meaningful results could be found in parts 2 and 4 which show high user *Satisfaction* levels for pulling in orthogonal direction indicated by the small box-plots that mostly lay down in the agreement ranges. Pulling the PullBand on the bottom area had the highest *Satisfaction* results in the whole pulling conditions. Fourteen out of sixteen users either agreed or strongly agreed with this condition to be the best among all the other conditions. Following that, pulling

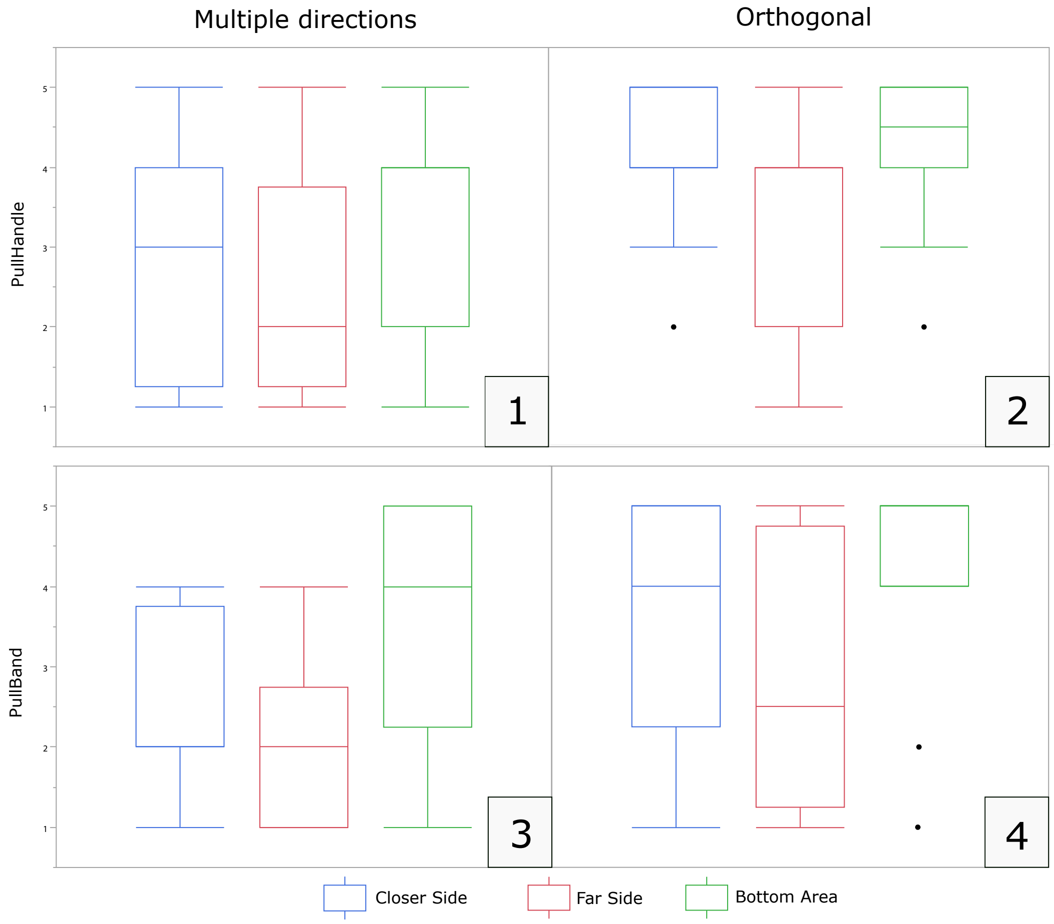


Figure 6.2: Pulling *Satisfaction Levels* on each of the interaction locations: pulling the PullHandle in multiple directions (1)/ one direction (2), pulling the PullBand in multiple directions (3)/ one direction(4). Pulling the PullBand on the bottom area had the best results.

Pulling in multiple directions was not preferred.

the PullHandle had the second best results on both of the bottom area and the closer side with 9 users out of 16 answering with agree or strongly agree.

The results also suggest that we could disregard pulling in multiple directions as the user feedback was spread out (part 1, 3) with tall box plots which emphasize users disagreement over one opinion. In the same manner, users disagreement could be noted with pulling on the far side in all the different conditions .

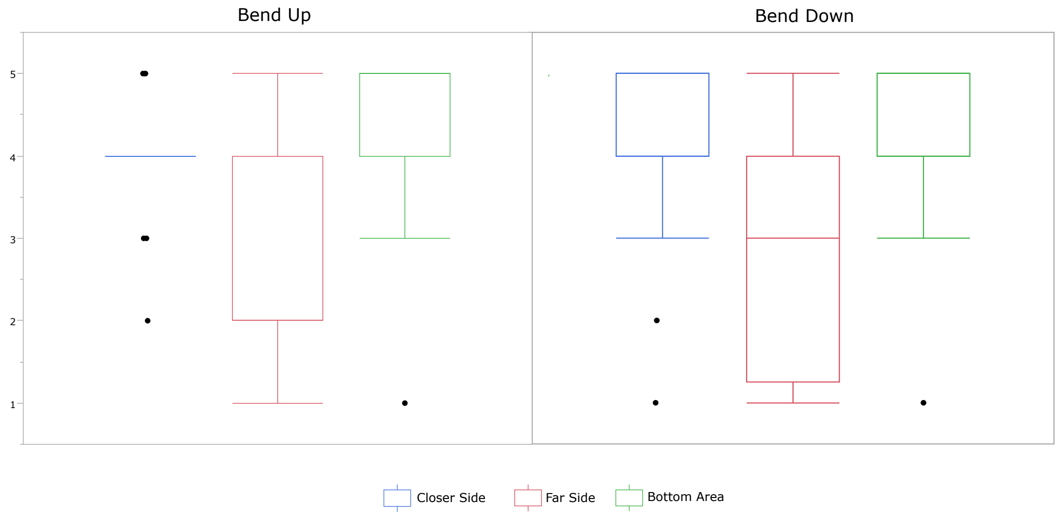


Figure 6.3: Bending *Satisfaction Levels* on each of the interaction locations. Far side was the least preferred location to facilitate bending.

6.5.2 Bending

Figure 6.3 shows that bending direction up or down did not have any effect on users *Satisfaction*. The Figure also shows that closer side and bottom area are much more preferable than the far side for interaction. In general, 75 percent of the users answered agree or strongly agree on the closer side compared to only 25 percent, on the far side.The bottom area had very similar results to the closer side.

The far side was also the least preferable location for bending

6.5.3 Twisting

The flexible strap's bending direction, to the top or to the bottom, did not have any effect on users' *Satisfaction*. Therefore we show the results only for FlexibleStraps that are bendable towards the top. Similar to previous results, Figure 6.4 shows that the far side preformed the worst and then comes the closer side. The bottom side has the best results.

The far side was also the worst to utilize twisting.

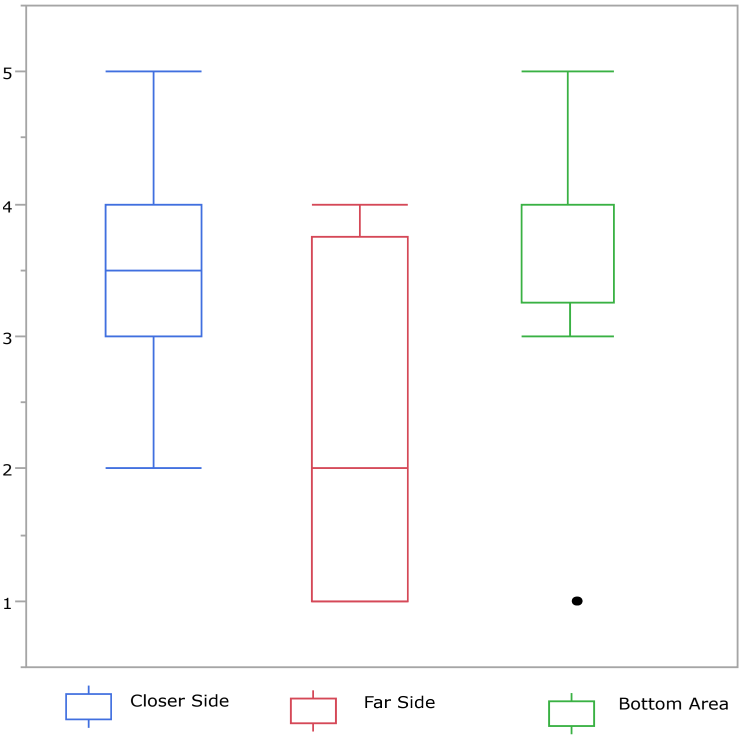


Figure 6.4: Twisting *Satisfaction Levels* on the different interaction locations. The bottom area was the most preferable

6.6 Discussion and Implications

User comments conveyed strong preference for pulling over bending and twisting.

During the study, we asked participants to comment on their most preferred technique which facilitates the act of deformation on the band. User feedback showed that pulling comes first, bending in second place, and twisting was the least preferred. Users comments indicated that pulling gave a better feel of control than the other techniques. They could see bending being used only in a snapping manner, where the strap is snapped quickly as a discrete or binary interaction input. Bending the strap in a continuous manner was much less preferable. In addition, twisting was not desired in any scenario. Users commented that twisting felt unintuitive, difficult, and resulted in an uncomfortable hand posture as it is required to firstly bend the strap and then to twist it.

Having identified puling as the most preferred technique for interaction, we will focus our discussion on analyzing the results for our pulling prototypes. We mainly presented two prototypes which have two different pulling affordances, the PullHandle and the PullBand. The PullHandle prototype offered free ended strings with a magnet as a signifier. Users can interact with the string by grasping and pulling the magnet. Our results indicate that this pulling affordance is flawed by the use of the magnets, which were supposed to be an improvement over the previous iteration. Users comments regarding their general dissatisfaction with magnet concepts referred to:

The PullHandle prototyped was flawed by many issues related to the magnet usage.

- Difficulty in detaching the magnet in order to pull the string.
- Disturbance due to the snapping sound coming after the magnet snaps back to its original position.
- No point of using the magnet as they do not always snap back to their position after pulling, leaving the cord hanging.
- Fear of the magnets due to the strong magnet's surface that could break or scratch the watch face when snapping back.

These results for the PullHandle prototype indicate many problems that need to be solved. Moving on to the PullBand, in which users slide their finger under the string and then pull, most users reported to be satisfied with the affordance that this prototype provides. However, users also pointed out some minimal problems. The string is too loose and too thin, making it not only annoying to be worn but also difficult to be found eyes-free.

PullBand was the most preferred prototype.

We finally asked our users whether they like the idea of pulling in multiple directions to control the band or they prefer to pull only in one direction. Users answered that pulling in one direction was much more preferred than pulling in multiple directions with both prototypes. Users commented that pulling in multiple directions was counterintuitive and awkward to be done. These comments line up with the results from the questionnaire.

Pulling in one direction is preferred over multiple directions.

The altered design utilizes a stretchable watch band that is in direct contact with the user's skin.

All in all, pulling the PullBand at the bottom area in orthogonal direction is the most preferred condition in this user study. However, this prototype had some problems related to how thin and loose it is. To fix that, we tested in an informal manner having a thicker stretchable band which acts as the standard band of the watch. Having this option allows for direct contact with the user skin, and consequently results in utilizing the user skin to act as a reference point for the user, telling him where to do his interaction without the need to look. In addition, we believe the direct contact with the skin would provide a better feel of control as the tactile feedback helps not only in finding the band but also in providing the feel of the amount of stretching force that the user is applying.

The next chapter will go through the hardware and software design and implementation details that supports this technique of interaction along with a final user study which evaluates the user performance using this technique for in-motion and eyes-free interaction.

Chapter 7

PullBand: Implementation

PullBand design came after extensive research and iterative design process.

In Chapter 3, we referred to the most common interaction problems with the smartwatch including screen occlusion and the fat finger problems. We highlighted the inability of the current smartwatch of supporting in-motion interaction despite its common usage. In Chapter 4, we presented how the physical tactile feedback, provided by the act of deformation, could be used to design interaction possibilities that support eyes-free and in-motion interaction. Subsequently, Chapter 5 showed the iterative design process of our low fidelity prototypes which support different kinds of deformation acts including bending, twisting, and pulling the wristband. The first study, in Chapter 6, showed that interacting with Pullband prototype at the bottom was the most preferable option among the other options and prototypes which we explored. It also suggested modifying the band design to be in direct contact with the user's wrist, making it more practical to be worn. This modified design also provides a tactile feedback, on the wearing hand skin, as the pulling force is being applied.

This chapter presents the implementation details of Pullband.

This chapter, in addition, focuses on the hardware and software implementation details of Pullband. In this manner, we indicate the reasons behind choosing a force sensor to measure pulling. We present the set-up we used to utilize the inclusion of the force sensor on the back of Apple

Watch. This novel sensor casing allowed us to detect the maximum range of the sensor's input along with the ability to recognize Quick Release events [Ramos et al., 2004]. The casing also ensures nullifying any possible noise which could affect the interaction. The chapter concludes with briefly referring to the set-up we used in transmitting the sensor's output to Apple Watch and iPhone through BLE development bored from [RedBear].

7.1 Utilizing Pulling for Target Selection

Pulling could be utilized to perform micro interactions from discrete nature.

We have seen in Section 2.2 how the smartwatch is commonly used to perform micro interactions from a discrete nature such as checking notifications, playing music, and starting or stopping a workout. The music application, for instance, provides a list of discrete commands, which the user can choose from: play/stop, play next song, and play the previous song. We could utilize pulling to scroll through a menu of discrete commands and select the desired target.

Using pulling, the user could scroll through a menu of discrete items.

The user can confirm his selection by quickly releasing the band.

We could map the pulling force to discrete menu items by splitting the range of the pulling force into separate levels that correspond to the menu items. Moreover, to select the target, we mainly thought of using Dwell Time or Quick Release [Ramos et al., 2004]. Dwell Time works by holding the band on the desired target in order to confirm the selection. However, this selection technique requires time, precision, and accuracy. The user has to maintain his pulling force for a short while until the target gets selected. Hence, this technique slows down the interaction. It is also difficult to be done in-motion where the user is walking or running. Another possibility for selection is using the concept of Quick Release. Once the user decided which target he wants to select, he can quickly release the band to confirm his selection, resulting in a faster interaction process. Quick release is a common technique that has been used for selecting targets using pressure input on the smartphone [Ramos et al., 2004].

In addition, we aim to find the right sensors which are able to capture Quick Release events on the band. Equally important, the sensor should be able to measure pulling force in order to map it to discrete menu items. The sensing mechanism should be robust, stable, precise, and responsive to support these requirements. In this manner, the following section presents our investigation to find a suitable sensor for our Pullband prototype.

With PullBand, we aim to support detecting Quick Release and utilizing the level of the pulling force.

7.1.1 Considering Stretch Sensors

As we were looking for a possibility to implement our pulling technique, we used the Adafruit sensor that we have discussed earlier in Section 5.1.1 where our preliminary tests showed them to be the most reliable among the other commercial and stitch-based stretch sensor types.

To test whether these sensors are able to support quick releasing and continuous input, we included a short piece of the sensor that matches the length of the wristband in our prototype and tested its behavior. Results showed that a significantly smaller output range of values compared to the range we got in the preliminary testing. Testing shows, that the shorter the piece of sensor that is used, the smaller the range of output; Therefore, the smaller the amount of utilized pulling force. Furthermore, the sensor showed a high level of instability with its values over time which makes it unreliable to use in the long term. Last by not least, the sensor makes it difficult to support the detection of Quick Release events as it needs extra time to shrink back to its original length once the pulling is stopped. All in all, due to the limitations of this stretch sensor, we looked for other sensing possibilities to implement our prototype.

The stretch sensor from Adafruit failed to meet the requirements needed to support Quick Release and utilizing the level of the pulling force.

7.1.2 Force Sensors

Force sensors have been used to detect pressure force level with high precision and very accurate and robust measurements [Ramos et al., 2004]. Moreover, they have the ability to go back quickly to the zero state once the force is

Figure 7.1: The Hook-and-loop band design taken from Apple watch sport band design. This design makes it easy for the band to fit most wrist sizes, keeping an affordance for pulling at the bottom of the band.

stopped being applied, which allows capturing Quick Release events. In order to utilize using force sensors to detect pulling, we thought of placing the force sensor on the opposite side of the pulling force. In addition, we placed an FSR 402 short' from Interlink Electronics on the back side of the watch face. When the wristband is pulled at the bottom, the watch face is pushed into the user's skin creating force value that is measurable by the force sensor. According to our knowledge, this concept is novel and have not been implemented before. Please note that placing the force sensor on the back of the watch has a limitation concerning the ability to capture heart rate with the heart sensor included in the back of the watch.

The force sensor is used to detect pulling by placing it on the back side of the watch face.

In addition, we designed a stretchable band which follows the hook-and-loop design by Apple watch loop band (Figure 7.1). This design makes it possible to be worn on a wide wrist sizes [145–220] mm. This allows it to able to be worn by males and females. This band design also facilitates stretching at the bottom and quick and easy adjustment.

7.2 Novel Set-Up to Detect Pulling and Quick Release

After testing placing the force sensor on the back side of the watch face we noticed an unavoidable amount of noise which could be explained by the following:

- The direct contact between the sensor surface and the user's wrist results in force detection and unavoidable noise.
- The backside of the smartwatch (Apple Watch Series 2), has a circle convex bump which creates a slight convexity in the sensor surface resulting in noise.

Placing the sensor on the back side of the watch suffered from noise and inconsistent readings.

We needed to think of a solution to eliminate or, at least, minimize the noise. A commonly used procedure is to provide a predefined threshold. However, in our case, the noise was reaching a high level of readings and the threshold needed to be also high which leads to loss of a wide range of the output. Alternatively, we used a foam layer to separate the back watch surface, where the sensor is located, from the user skin. This sandwiched foam layer shrinks once the user starts pulling the band allowing the sensor to get in contact with the user's skin and thus detect the force. However, although this approach eliminated the unwanted noise, it still was not able to assure a robust output of the sensor. This goes back to the reason that the user's wrist is not flat and the contact point between the sensor and the user skin could be different between different pulling attempts. Therefore, we designed a three-layered casing that includes the force sensor. The casing consists of two solid parts and another foam part sandwiched in the middle between them. We printed a three dimensional model of the solid parts using Polylactic Acid (PLA) material and infill of 100 percent. In the following, we provide the specifications for the sensor casing (as seen in Figure 7.2):

The casing set-up consists of a foam layer, used to omit noise, sandwiched between two solid parts that, together, ensure consistent readings.

- The first solid part is placed on the back surface of the watch and has dimensions of 37 mm in width, 42

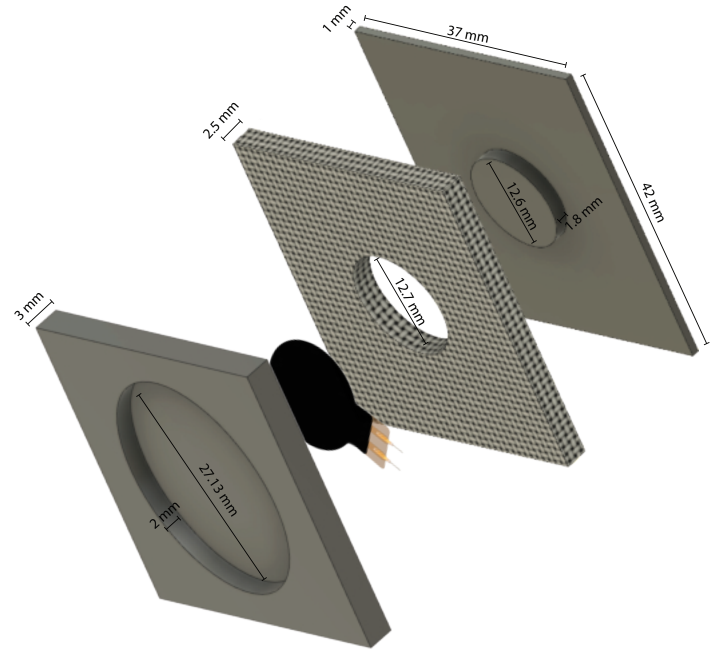


Figure 7.2: The three-layered force sensor casing: the sandwiched foam layer in between the two PLA layers assures omitting any unwanted noise, consistent readings, and the capturing of Quick Release events.

The first solid part is used to provide a flat base for the sensor.

mm in length, and 3 mm in height. It also has a circle concave of a diameter of 27.13 mm, and height of 2 mm (Figure 7.2). The main objective of this layer is to provide the sensor with a flat base and eliminate the noise caused by the convex surface of the watch back face.

The foam layer assures the isolation of the sensor when no force is applied.

- The sandwiched foam layer is cut by a laser cutter and have the dimensions of 37 in width, 42 mm in length, and 2 mm in height. We created a circular hole in the center of this layer with a diameter that equals the active area of the force sensor 12.7 mm. The foam material is flexible which allows this material to shrink, once the user starts pulling, bringing the force sensor closer to the third solid part allowing the detection of force to start happening(Figure 7.3).
- The third part is solid and consists of a flat surface of 37 mm in width, 42 mm in length, and 1 mm in height.

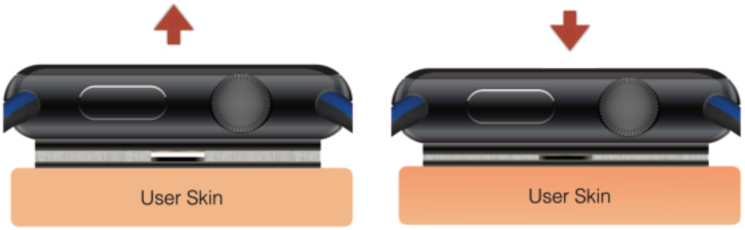


Figure 7.3: The concept in action: once pulling has started, the watch face gets pushed towards the user skin causing the foam layer to shrink and the sensor to start reading values.

It has, in the center of it, a small cylinder ledge 1.8 mm in height. This small cylinder ledge goes through the sandwiched hole in the foam but its height does not allow it to be in contact with the force sensor unless pulling has started(Figure 7.3). Another advantage of using this cylinder ledge is to focus the force at the same space of the sensor's sensitive area, every pulling attempt, which helps to first get the full range of values and secondly to have consistent output results regardless of the shape of the wrist or the contact point facing the sensor.

The cylinder ledge assures distributing the force on the sensitive area of the sensor.

Having the previous casing assures noise occlusion, and allows force detection only if an adequate amount of force was applied. Moreover, it provides a robust and consistent detection as it assures the force to be equally distributed on the sensitive area of the sensor. The printed version of this casing could be seen in Figure 7.4.

7.3 Capturing Quick Release

To capture Quick Release events, we were inspired by the study results of Corsten et al. [2017] and the human processor model that Card and Moran introduced. According to Card and Moran, any action performed by a user con-

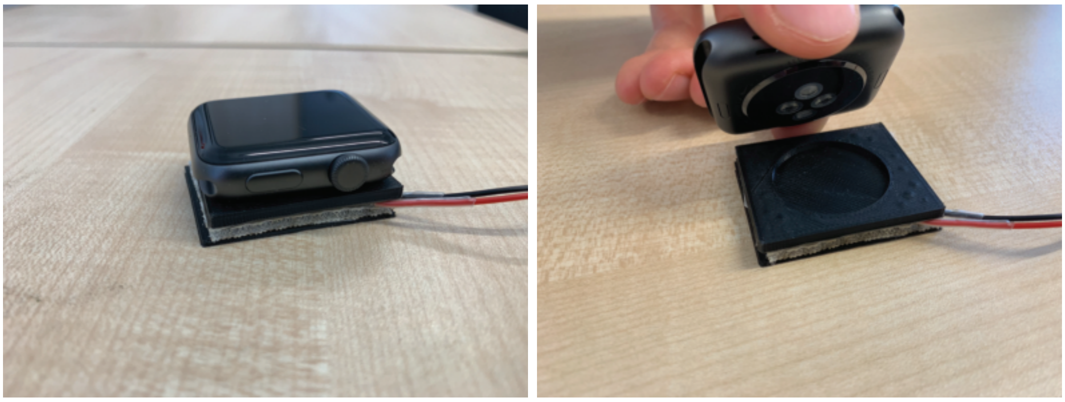


Figure 7.4: The printed version of the sensor casing.

We capture Quick Release events between [239..250] ms before the force reaches zero.

sists of three main steps where each of them has a specific amount of time to be completed. These steps are (1) perception (100 ms), (2) cognition (70 ms), and (3) motion (70 ms). In the context of applying Quick Release for selection, we map the previous steps to context-specific steps as follows: (1) the user makes sure which item is currently selected, (2) the user decides to confirm his selection, and (3) the user quickly snaps out his finger releasing the band at the desirable level of selection. Subsequently, the Quick Release event should be captured around 240 ms before the force reaches zero and the sensor is not capturing any more pulling force. We informally tested our prototype with multiple users. As assumed, our test showed that correct selections always fall in [239..250] ms before the readings reach its zero value.

7.4 Linearizing the Sensor's Output

The output of the force sensors is not linear i.e., the relation between the input, the force applied, and the output is not linear rather closer to be logarithmic. We need a linear sensor output as we wanted to map the pulling force to discrete menu items by diving the pulling range into equal parts that correspond to discrete menu items. Hence, we calibrated our sensors by a placing a small piece of rubber in the center of the sensor with a diameter of 12 mm and putting an empty glass on the top of it. We start adding water to the glass to simulate different weights. We executed the same procedure on three different force sensors and then calculated the mean results to formulate the transfer function. We also repeated the measures for each sensor three times to make sure of the values. Our tests showed the maximum output of the sensor resulted with (4.4 N) of force that is being applied. The transfer function is reported below:

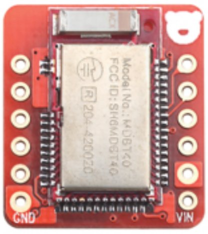
The linear transfer function results in a linear range of output that could be mapped to a discrete menu list.

$$f(sensorValue) = 15.281 \times e^{0.0042 \times (sensorValue)}$$

Moreover, we tested our function on two other sensors out of the group that we used to calibrate our sensors. However, it should be emphasized that evaluating the transfer-function is beyond the scope of this thesis. With that, the sensor output is now linear and ready to be used in our user study.

7.5 Transmitting the Sensor's Output to Apple Watch and iPhone

We connect the sensor to a BLE nano from RedBear, which is one of the smallest Bluetooth Low Energy (BLE) development boards in the market. The sensor is connected to 0-3.3V analog input which is mapped to 0-1023 digital units using a pull-down resistor 10 kilo ohm. We send the data from the BLE nano to Apple Watch Series 2 through Bluetooth using the Core Bluetooth framework, which allows a communication between Apple Watch and Bluetooth 4.0 low-energy devices [Apple, 2018].



The BLE nano from RedBear

In order to test our set-up we developed an application to capture the input of the sensor and to trigger haptic feedback once the user starts pulling. The application worked as expected once it is active and on screen. However, it is worth mentioning that using WatchKit does not allow playing audio feedback through the watch speakers. As an alternative solution to test implementing audio feedback, we

Using watchOS 4, we suffered to support audio feedback using the watch speakers.

connected the BLE nano to iPhone 5s, making it possible to trigger audio feedback.

This chapter provided the implementation details of our high fidelity prototype, PullBand. Next chapter provides the evaluation details of PullBand for menu selection tasks in-motion and eyes-free, along with the results and the findings. Please be aware that the decision of which type of additional feedback, whether it is auditory or haptic, will be discussed in the next chapter.

Chapter 8

PullBand: Evaluation

We evaluate using the PullBand prototype to select menu items in different motion levels.

In the previous chapter, we presented our final prototype "PullBand" and indicated the implementation details which utilized pulling any stretchable smartwatch band to perform micro interactions from discrete nature. We claim that by using PullBand, the user could scroll through a menu and select a discrete command on the go for in-motion and eye-free interaction. In this chapter, we evaluate the user performance using our selection technique, against different discrete menu sizes, motion levels, and pulling force levels. We present the study design, including the hypotheses and the essential design decisions we took to make our prototype suitable for the purpose of this user study. We follow up by showing the utilized task, the independent and the *dependent variables*, along with the experiment's target group. Last but not least, we indicate the decisions we took in the process of collecting and filtering the data. The chapter concludes with the results, of the statistical analysis, along with a discussion of their interpretation.

8.1 Hypotheses

Throughout the study, we examine the following hypotheses while the user is in-motion and is not looking at the watch (stated in null form, i.e., expected to be rejected):

H1 In-motion interaction will decrease performance, user's performance is dependent on the motion level of the user.

H2 Menu size will have no effect on user's performance, i.e., user's performance is independent of discrete menu size.

H3 Pulling force level will have no effect on user's performance, i.e., user's performance is independent of pulling force applied.

8.2 Feedback Design Decisions

To support eyes-free and in-motion selection, audio or haptic feedback will be provided.

Our design could provide haptic or auditory feedback as an alternative to the visual feedback in order to support eyes-free and in-motion interaction. Eyes-free interaction devices tend to specify predefined points in the input space where the feedback is triggered. This feedback tends to tell the user that a transition has occurred from the previous state to the current state. However, this alternative feedback should be also kept to the minimum in order to not consume much attention from the users. For the purpose of this user study, we decided to use audio feedback instead of haptic feedback. In this section, we will go through the reasons behind our decision.

8.2.1 Amount of Information and Period of Feedback

It must be understood that there is a trade off between the speed of the feedback and the amount of information that it can carry [Oakley and Park, 2007]. In general, audio feedback can carry more information than haptic feedback, which is only able to convey short meaningful signals that notify the user about an event. However, when using these short signals, we have to consider the amount of effort needed to interpret the feedback especially eyes-free and in-motion, i.e., It is much harder to interpret the meaning of short tune messages than to understand a full speech command. Imagine a scenario where the user is asked to select the fourth menu item in a seven menu item list. In case of using short signal as feedback, it would take a relatively long time to select since the user needs to go through the menu slowly until he reaches the fourth tune. On the other hand, using the voice feedback that conveys an ordered list, the user might estimate the amount of pulling force he has to apply in order to jump right away to the wanted level.

Haptic feedback is faster than auditory feedback but harder to interrupt.

Using an ordered audio list helps to perform jumps with the estimated pulling force.

In this user study, we ask the user to scroll through an ordered menu to find a specific menu item. The user has the option to pull and reach the item straight forward or he could scroll the menu item by item until he reaches the target. We needed specific and clear audio commands that convey order. Therefore, we introduced spoken items that consist of only one syllable which is a number between one and seven. For instance, if the user hears the voice command ”one”, he would define a plan to scroll to the first menu item whereas if he hears the last menu item, then he is able to apply maximum force.

We implement an audio menu with spoken numbers.

It is important to remind the reader that the time needed for selection is heavily affected by the audio items’ feedback durations which add up much time to actual *Selection Time*. For example, one audio feedback is in the range of [400..600] ms and in the case the user hears 2 audio menu items before he selects, this could sum up to at least 1200 ms of just audio listing time.

An audio item takes [400..600] ms to be fully spoken.

8.2.2 Speed of Triggering Feedback

In general, feedback has to be triggered very quickly and without any noticeable delay, since the feedback is needed by the user to be confident of his actions [Oakley and Park, 2007]. In our prototype, we choose to trigger discrete audio feedback on predefined points to inform the user that he is currently in a new menu item. The feedback, however, is only triggered if the user holds on one level for a short amount of time. For example, if the user is trying to select the last possible level, he would apply the maximum pulling force to reach it. Therefore, the user would only expect to hear the feedback of the last level and not any other value that he passed especially because audio feedback takes time to be fully spoken. Therefore we chose to introduce a short delay in triggering the audio feedback of 100 ms. This guarantees that the feedback would only be triggered if the user stays inside one level for more than 100 ms. This will also prevent any spoken feedback when the user is just passing by the values.

The audio feedback is triggered with a short delay of 100 ms for smooth feedback experience.

As we indicated difficulties with triggering audio feedback on Apple Watch Series 2 in Section 7.5 using the watch speaker, we chose to trigger the audio feedback using iPhone 5s speakers. For that, the sensor output is connected to the iPhone through BLE nano interface. Having the feedback through speakers was important in the user study as we wanted both the inspector and the participant to hear the audio item. However, if the technique would be utilized in every-day life scenario we think that triggering the feedback through headphones would be better as the user would get the feedback directly in his ears.

We trigger audio feedback using the iPhone 5s speakers.

8.3 Setting

Referring to the overall surroundings, participants are asked to perform the interaction either while standing or while walking on a treadmill. Interacting with the band while walking on a treadmill simulates an in-motion interaction scenario as the user cognitive load is split between performing the interaction and focusing on keeping his speed stable on the treadmill. In this user study, we decided to use a treadmill instead of figure eight, which we used in the first study, in order to make sure that the user speed is stable and consistent over all the study. Moreover, walking on a treadmill simulates an in-motion interaction scenario where the user's cognitive load resolves around the concern of keeping his current speed and not falling out of the machine. We offered breaks every time the user had to start a new motion condition. This helps to give the participant a chance to recover so that he does not get fatigued while performing the task. It should be mentioned that considerable care had to be taken to prevent participants from looking at the band while performing the task in order to ensure eyes-free interaction.

The user is asked to walk on a treadmill to simulate in-motion interaction.

8.4 Task

8.4.1 Main Task

To evaluate user performance, using PullBand, we adapted a menu selection task to assess differences in performance of different motion levels, menu sizes, and pulling force levels. Consequently, as we intend to evaluate eye free interaction, we present an auditory stimulus to our participants using an iPhone 5s held by the study inspector. The user hears the audio stimulus which asks him to navigate to specific discrete menu item, out of a predefined range, and then to confirm his selection as quickly and accurately as possible. Audio feedback of the current menu item is provided once the user remains on a menu item for more than 100 ms. The selection is made as soon as the user release the band. Once the selection has been made, a natural sound is provided as a feedback to reflect the success or failure of the task. The natural sound is adopted from Apple design guidelines for haptic feedback on the smartwatch [Apple, 2018]. In the following, we provide the detailed interaction steps:

The user is asked to select an audio menu item as accurately and quickly as possibly.

1. An auditory stimulus is spoken by a female voice, Siri Samantha Apple [2018]. The stimulus has to be fully spoken before the user can start his interaction.

Stimulus is spoken using a female voice.

2. The user slides his finger under the bottom area of the band.

The feedback is spoken using a male voice to differentiate it from the stimulus.

3. After the user places his finger under the band, he is able to navigate to the intended audio target by applying pulling force, as quickly and accurately as possible. The discrete current target is automatically spoken whenever the user rests his pulling force on it for more than 100 ms. The feedback is spoken by a male voice to differentiate the feedback from the stimulus and avoid any confusion. The feedback has to be fully spoken before the user can select the target.

4. When the user feels confident about his selection and wants to confirm, he quickly release the band. The user's selection is confirmed using a short natural audio feedback that confirm success or failure.

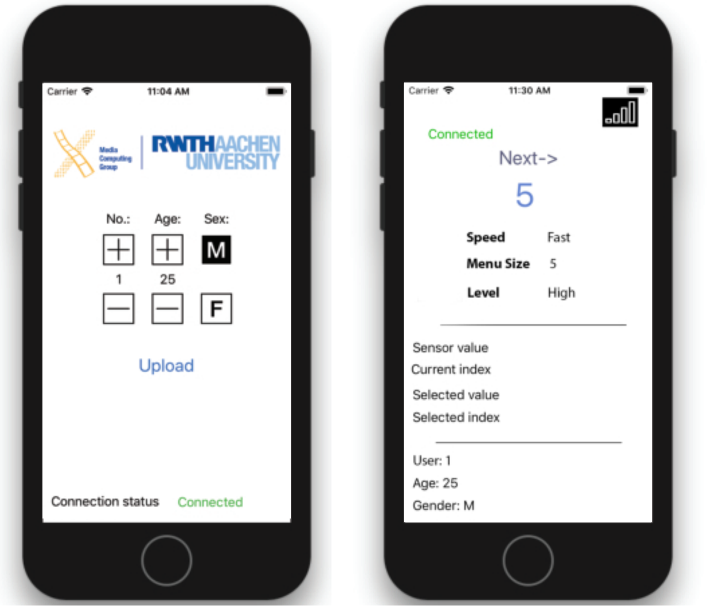
5. The user is asked to put his hands apart so he can start the next trial from a natural position of interaction. This helps to capture the *Preparation Time* needed for the user in order to start the interaction. It also ensures getting consistent results over the whole set of commands.

6. A new trial is started, manually by the inspector, using the next button on the iPhone app shown in Figure 8.1, once the user puts his hands apart.

The inspector guides the user throughout the study.

7. Considering that tasks consist of three different menu sizes, the user has to be informed once the menu size changes. The size of the menu is always shown on the screen of the iPhone that is carried by the inspector seen in Figure 8.1. Subsequently, the inspector has to inform the user every time the menu size changes. It should be mentioned that we chose to show this information only to the inspector throughout the study to keep the user's visual cognitive load to the minimum.

8. Tasks are performed while the user is in different motion levels. After the user performs all the tasks in each motion level, he is asked to sit down and take a short break of at least 2 minutes.



No.: 1
Age: 25
Sex: M
Connection status Connected

Connected
Speed Fast
Menu Size 5
Level High
Sensor value
Current index
Selected value
Selected index
User: 1
Age: 25
Gender: M

Figure 8.1: iPhone app used by the inspector of the study: provides automated study set-up depending on the participant's number. The participant only hears the audio feedback provided by the app.

After the user finishes all the tasks, we asked him to take off the smartwatch and then handed him a general questionnaire about the study.

8.4.2 Visual Exploratory Task

The visual feedback provides an exploratory learning mode.

When designing eyes-free interface we should take into account how easy it is for novice users to learn the system. Graphical user interfaces provide an exploratory mode of learning i.e, the user can learn how to interact with the graphical user interface by exploring it [Brewster et al., 2003]. For example, with menu selection task on the smartwatch, through visual feedback, the user can immediately see the feedback on the screen about his position in the list (at which menu item) and how fast and which direction he is scrolling.

Figure 8.2: Visual Exploratory Watch App with a menu selection task. The user navigate through the menu in order to select the highlighted target in orange. In the success scenario at the top, the user select the right target. In the scenario at the bottom, the user selects a wrong menu item.

Users familiarize themselves with the technique using our visual exploratory app.

We decided to include a visual exploratory watch app that allows users to try the selection mechanism with a visual feedback. In this manner, users have the opportunity to familiarize themselves with both scrolling through the contents and applying the Quick Release selection mechanism. In addition, participants can develop a feeling for how much force is required to navigate to the targeted location. Therefore, we developed an application on Apple Watch where we designed a menu of four items as seen in Figure 8.2. Please be aware that we chose a menu size that does not match any of the menu sizes which we will test in the study in order to minimize any learning effects for specific menu size over the others.

The confirmation feedback is coupled with the same failure and success sounds used in the user study.

The targeted menu item is highlighted in bright orange color and the currently selected item is in dark blue color. Once the user confirm his selection, the chosen item is highlighted either in green for a successful selection or in red for failed selection. The visual feedback of this confirmation is also accompanied with the same natural sounds of failure and success which we use in our audio feedback during the study. This helps the user to learn to differentiate the failure and the success sounds in order to minimize learning effects during the study.

Having referred to main design decisions that we had to make along with the study task, the settings, and the visual exploratory watch application, we are going to move on to talk about the design details of the user study including the independent and the *dependent variables*.

8.5 Design

8.5.1 Independent Variables

Throughout the study we control the following conditions:

Motion Level

Motion level places different amounts of cognitive load on the user.

We test whether pulling the band for selection tasks could be performed in-motion with the same performance as while standing. In this regard, we distinguish between three different motion levels: standing; slow walking, with speed of 2.5 km per hour; and speed walking, with speed of 5.0 km per hour. We test two different walking speeds to simulate different cognitive load scenarios. The first walking speed is slow where minimal cognitive is occupied by the motion level. The second walking speed of 5.0 km per hour simulates the average walking speed for pedestrians on a crosswalk [Aspelin, 2005], and has a higher amount of occupied cognitive load.

Menu Size

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

We control menu size to specify the number of items among which predefined targets are chosen. Note that the amount of distinguishable levels is crucial, since it specifies the achievable bandwidth when using pulling as an interaction modality. Vogl et al. [2017] used menu sizes with four, five, and six item in his user study, but he did not actually test the performance or the target selection. We decided to evaluate menu sizes of three, five, and seven items. We chose these sizes with 3 items as we rely on the fact that various watch applications tend to have 3 common commands which are easily accessible on the go. For example, with the telephone application, the user could answer a phone call, dismiss it, or dismiss it with a message. Similar tasks could be achieved with the incoming notification. With the workout application, the user could start a workout, stop it, or pause it. Another example is the music application where most three common actions are play/pause, play next song, and play previous. Having a larger set of items such as 5 and 7 will also allow support a larger set of commands. For example, with 5 commands in the music application we can add turning the sound up and down.

Pulling Force Level

We evaluate three pulling force levels.

We test three pulling force targets with low force, middle amount of force, and high force. We test the same force targets across the three menu sizes using a method adopted from Ramos et al. [2004] which helps in comparing each force level across the different menu sizes.

We do not use the full range of sensor values to keep the technique comfortable to use.

We choose the specific forces to be the center of our levels: low 0.73 N, middle 1.5 N and high 2.8 N. Please be aware that although our tests for the force sensor (presented in Section 7.4) showed that the maximum measured force is 4.4 N, we did not use, in this user study, the full range of input. We chose to make it easier for the users to reach the maximum pulling level keeping the technique practical and comfortable to use. In this manner, we used 2.8 N to be the center of our highest force level. We also introduce a small threshold with the value 0.3 N.

The low and high force levels are mapped into middle items in menu size seven.

The force levels fall into the first, the second, and the third menu items in menu size three; in the first, the third, and the fifth on menu size five; and in the second, the forth, and the sixth with menu size seven. Please note how the low and the high force targets, in the menus with three and the five items, are mapped into the first and last possible items respectively. On the other hand, on the menu size with seven items, they are mapped into middle items instead. We will make sure, in our analysis, to take these differences into account, if any significant difference was found.

8.5.2 Dependent Variables

By controlling motion level, menu size, and pulling force level we ensure that participants perform the selection task under different conditions. To capture users' performance of the different tasks, the following *dependent variables* are calculated:

Preparation Time [seconds]

Represents the first *dependent variable* and is defined by the total time that the user needs to plan his interaction, and then starts pulling. Once the audio stimulus is fully heard, a stop watch is started. It is stopped once the user starts pulling the band. We think that measuring the *Preparation Time* is important to note whether the user needs more time to interact with the watch in-motion interaction scenario than the time he needs in a stationary mode. This could mainly differ as more cognitive load is occupied while the user is in-motion than when he is in stationary mode. Longer *Preparation Time* could also indicate difficulty in the respected condition as users tend to take more time planning their interaction the more difficult they perceive it to be.

Preparation time can convey information about difficulty in planning the interaction.

Selection Time [seconds]

This variable is defined by the total time that is required to acquire the right audio menu item, and to confirm the selection using Quick Release mechanism. A timer is also used to keep track of the time as soon as the *Preparation Time* ends, i.e, once the user starts pulling the band. The timer stops once the user releases the band and the selection is confirmed.

Selection time is measured since the user starts pulling until the target is confirmed.

Target Accuracy [true, false]

Target accuracy identifies the conditions where users preformed the best.

Indicates whether the user is able to select the intended target or not. In this manner, target-accuracy shows how many times the user made a wrong selection. It should be mentioned that we also consider selecting any value below the threshold a failed selection.

User-Satisfaction [5-point Likert-scale]

User's *Satisfaction* allows to assess qualitative data.

Last but not least, user *Satisfaction* provides users' personal opinion about using Quick Release as a selection mechanism with the band. We also ask the user about their *Satisfaction* levels concerning using the band in both stationary and in-motion interaction scenarios (Appendix B). Please note that satisfaction levels are qualitative variables and differs from all the other quantitative ones that we have previously introduced.

Having referred to the independent and the *dependent variables* of this user study, we can move on to describe the experimental design, including the number of conditions, and how counterbalancing is achieved.

8.5.3 Experimental Design

The study uses a within-subject design.

As we have done in the first user study, we also chose within-subject design in which each participant is presented with all of the conditions. Having within-subject design, we minimize possible biases that go back to individual differences; it also requires limited number of users compared to the between-subject design, which requires multiple separated groups of users. However, and as we mentioned in the first study, within-subject design also has some disadvantages which we have to pay attention to. We have to take care of carry-over effects, such as learning-effects as the user starts to learn how to use the system throughout the study. Moreover, we also have to consider that participants might get fatigued while performing the selection task, especially when they have to do it while in-motion. In order to minimize the disadvantages of within-subject design, we used counterbalancing, as well as sufficient breaks to recover.

It is worth mentioning that we have not implemented a total randomization for all the conditions. This is because total randomization means that the user has to keep alternating between different motion levels and different menus all the time. For example, the user has to do menu size three at fast speed then stops to do menu size five at slow speed and so on. This would results in a high possibility of causing confusion that would inevitably affects our results. Therefore, we split our study into three main motion level blocks; each speed block is split into three menu sizes sub-blocks. Both of the main and the sub-blocks are balanced using Latin square. In each of these sub-blocks, the user is presented with three force targets with three repetitions. Before each sub-block, three training trials are presented to the user, one for each of the three force targets. The user has always the option of repetition during the training trials. Once the user feels familiar with the new conditions he can start his study trials; The option of repetition during the study trials is given if the user, for example, accidentally pulls the band before he hears the audio voice message or needs to pause to take a rest.

The order of the blocks and the sub-blocks is balanced using Latin Square.

Before each menu size block, three training trials are presented to the user to familiarize himself with the new menu size.

Hence, each participant performs (3+3+3)*3*3 = 81 trials, and (3+3+3)*3 training trials yielding a total duration of (108*10 s) = 60 s = 30 min. We add to this the time needed for the pre-exploration of the visual feedback menu on Apple Watch, breaks durations, and the time needed for answering the short questionnaire. In total, the study takes approximately 45 minutes per participant.

The study took around 45 min per participant.

In the next section, we conclude the study design by defining the details of the targeted participants who took part in this user study.

8.5.4 Participants

Twelve participants, males and females, took part in the study.

Twelve users were recruited to participate. Participants were all right handed and wear the watch on the left wrist. All participants were students between 20 and 28 (M = 23.41). We recruited both male and female participants (9 males, 3 females) as our wristband was designed to suit both females and males with wrist sizes between 145 and 220 mm. The possibility of withdrawing is also given at every point in the study.

8.6 Data Collection

Collected data is stored in comma separated files.

Continuous input data is logged every 30 ms throughout the study.

Our evaluation requires capturing and storing accurate and continuous measures for the user performance and target selection. It is important to store our data in an accessible and structured format so it can be easily understandable and usable for later analysis. In this manner, we decided to use comma separated files (csv-files). We developed a mobile application on the iPhone to collect the data real time and store it during the course of study. For each participant, we store the data in two different files. The columns of the first csv-file are structured into context information, independent- and *dependent variables*. Context information shows the participant-ID and trial-ID, and the processing index with a timestamps. In addition, we allocate one column for task type which could be a regular or a training trial. This file also contains independent variables as motion level, force level, and menu size. Finally, measurements of the *dependent variables* are stored, including *Preparation Time*, *Selection Time*, as well as a Boolean value, indicating whether the task was successful. In addition, the second csv file contains continuous input data which is logged throughout the study. Values are logged every 30 ms. This data allow us to examine the patterns of selection of an intended menu item. Individual files can later be combined into a single csv file, containing all measurements categorized by the participant ID.

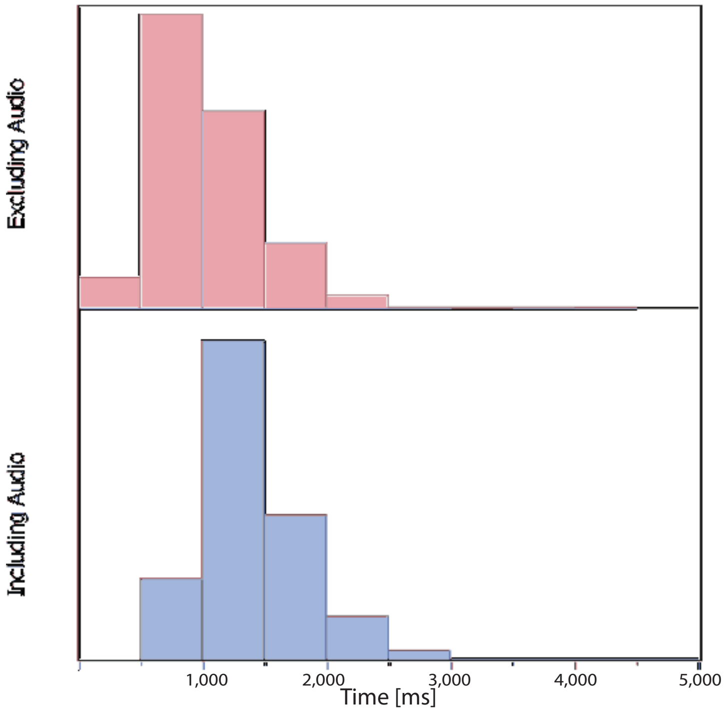


Figure 8.3: Adjusting Preparation Time: subtract the time needed to hear the audio stimulus from *Preparation Time*.

8.7 Post Processing of Data

Preparation Time Fix

Our software captures the time of each trial once the voice feedback starts. However, the user was asked to start his interaction only once the audio item is fully spoken. Therefore, we subtract the stimulus duration (500 ms) from the captured *Preparation Time*. Figure 8.3 shows the average *Preparation Time* before and after this subtraction.

We subtract the time needed to hear the audio stimulus.

Outliers

A total of 2 data points were removed since the value of their *Preparation Time* was too long (13 sec, 19 sec) compared

2 data points are excluded.

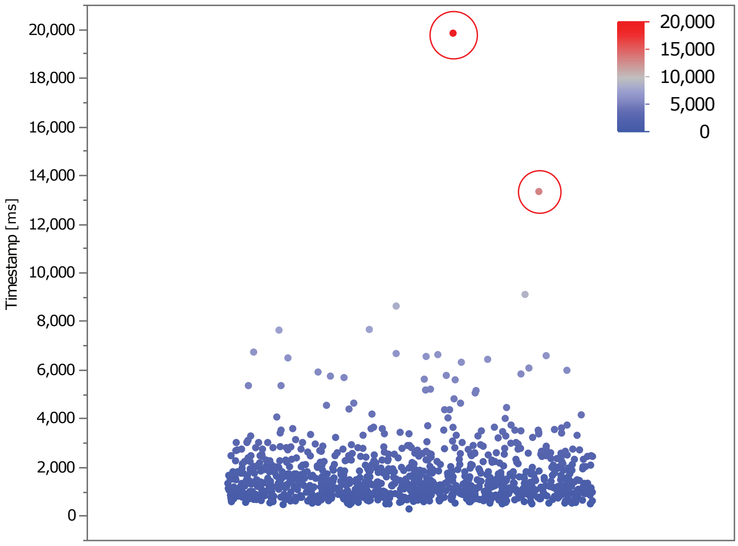


Figure 8.4: *Preparation time* for 974 data points. The 2 data points in red are excluded from the analysis along with their respective trials leaving 972 data points.

to all other trials. These excluded data points could be seen in Figure 8.4 in red. In these trials, users were still asking questions before performing the task. Therefore, these trials were not representative and we excluded them from the analysis.

8.8 Results

Results regarding *Preparation Time* and *Selection Time* are stated together.

Going back to our hypotheses (Section 8.1), the aim of this analysis is to determine whether H1, H2, ad H3 should be accepted or rejected and this section will be devoted to this purpose. As we stated in Section 8.5.2 the user performance is mainly measured by three *dependent variables*. The first variable is the time needed to start the interaction since hearing the voice command (*Preparation Time*); the second dependent variable is the time needed to select the specified menu item since pulling started (*Selection Time*); the third dependent variable is *Target Accuracy*.

Throughout the current sections, we start by stating the results regarding *Preparation Time* and *Selection Time* and then move on to state the regarding *Target Accuracy*. The section concludes with a discussion of the results to assess each of H1, H2, and H3. For both *Preparation Time* and *Selection Time*, the responses' means are illustrated along with error-bars representing 95% of confidence interval. To access the effects of the different levels in each of the *dependent variables*, we use the REML-method, i.e., Restricted Maximum Likelihood, and consider participants number as a random-factor [Wobbrock, 2011]. Please note that in order to be able to apply these tests on our data, we had to assure beforehand that it meets the requirements for normal distribution by applying logarithmic transformation on the both the *Preparation* and the *Selection Time*. Moreover, we applied Tukey HSD posthoc pairwise comparisons and we represented the results in tables; each level is presented in one row along with a letter symbol and the corresponding least mean square value. Only conditions which are not connected by the same letter are significantly different.

REML-method minimizes possible learning effects.

Tukey HSD posthoc pairwise comparisons results are stated in tables.

In addition, for measuring the significance in difference for *Target Accuracy*, we conducted McNemar test. For posthoc comparisons, we used the related samples Cochran's Q tests since the data was dichotomous. Success and error rates are also illustrated.

Regarding *Target Accuracy*, we ran McNemar and Cochran's Q tests.

8.8.1 Preparation Time & Selection Time

The following section will present the results of independent variables on time.

As illustrated in Figure 8.5, the analysis did not show any significant effect of motion level on both *Preparation Time* ($F_{2,954}$ = 1.57, p = 0.20), and *Selection Time* ($F_{2,954}$ = 2.52, p = 0.08).

Motion level had no significant main effect.

Menu size showed a significant effect on *Preparation Time* ($F_{2,954}$ = 52.61, $p < .0001$). Tukey HSD posthoc pairwise comparisons between different Menu sizes were all significant, i.e., *Preparation Time* increased with menu size. Signif-

Menu size had a significant effect on *Preparation Time*.

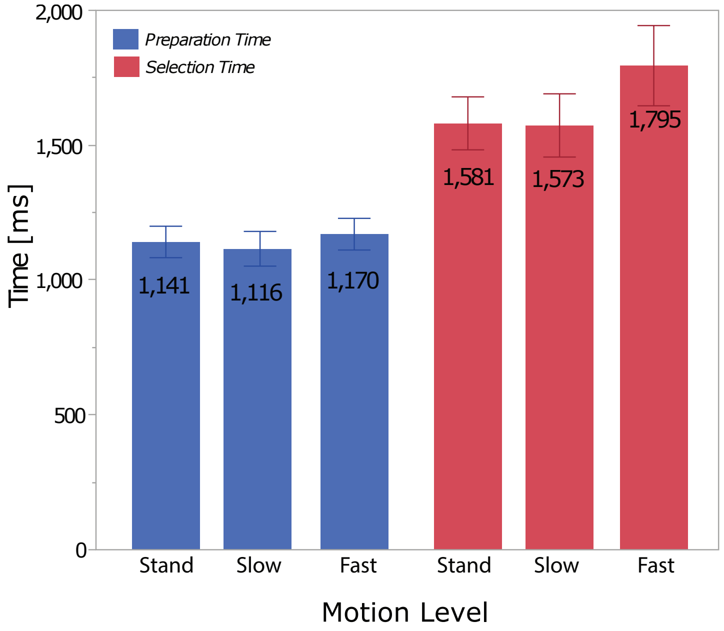


Figure 8.5: Means of *Preparation Time* and *Selection Time* [ms] according to motion level (Error bars denote 95% confidence intervals (CIs).

Participants were faster with menu size three.

icant difference between different menu sizes is illustrated in Table 8.1, where each menu size is referred to by a different letter symbol (A, B, or C). *Preparation Time* for the menu size with three menu items, referred to by the letter (A), was significantly shorter (LMS = 897.96 ms) than both *Preparation Time* for the five menu items , referred to by letter B, and the seven menu items, referred to by letter C. In addition, the table shows that the menu with five items also had a significantly shorter *Preparation Time* (LMS = 1,042.87 ms) than the time needed for the seven menu items menu (LMS = 1,203.80 ms). Figure 8.6 shows results regarding *Preparation Time*, along with error-bars representing 95% confidence intervals (CIs).

Menu size had a significant main-effect on *Selection Time*.

Similarly, our analysis showed a significant effect for menu size on *Selection Time* ($F_{2,954} = 201.44$, $p < .0001$). Tukey HSD post hoc pairwise comparisons for each menu size were all significant, i.e., *Selection Time* increased with menu size.

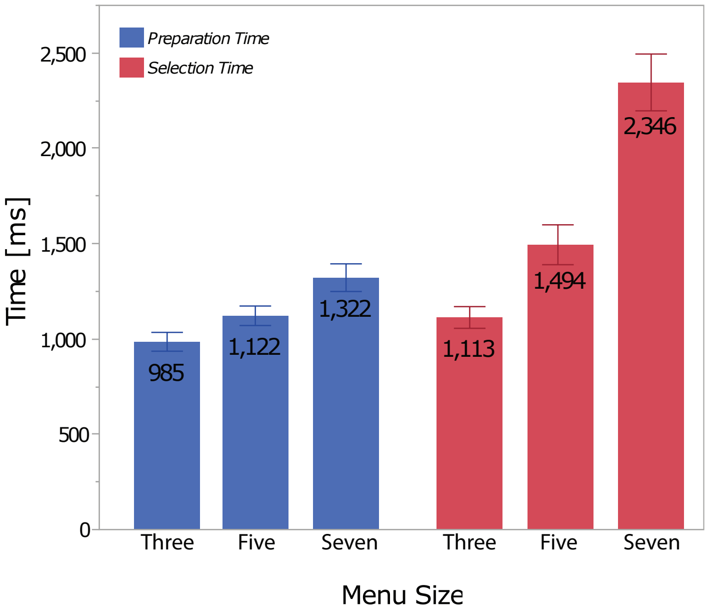


Figure 8.6: Means of *Preparation Time* and *Selection Time* [ms] according to menu size (Error bars denote 95% confidence intervals (CIs).

Table 8.1 reveals that the *Selection Time* needed for three menu items was significantly shorter than both the *Selection Time* for the five menu items and seven menu items(A,B,C). However, it is important to note that this difference was rather small between the menus with three menu items (LSM = 1,016.08 ms) and five menu items (LSM = 1,288.16 ms). On the other hand, the difference was rather big for the menu with seven menu items (LSM = 2,061.87 ms) as it needed much longer *Selection Time*. Figure 8.6 shows a bar chart illustration indicating the mean for both *Preparation Time* and *Selection Time*s along with 95% confidence intervals (CIs).

We captured an interaction effect of motion level on menu size 5.

We attract the reader attention to an interaction effect of menu sizes and motion level which we have encountered during our analysis as seen in table 8.2 by looking at the menu size with five items. The *Preparation Time* while walking fast (LMS = 1,171.85 ms) coded by letter (A), was sig-

	Menu Size				**Least Sq Mean**
Preparation Time	**7**	A			1,203.80
	5		B		1,042.87
	3			C	897.96
Selection Time	**7**	A			2,061.87
	5		B		1,288.16
	3			C	1,016.08

Table 8.1: Significant difference with the resulting *Preparation Time*, and *Selection Time* according to menu size. Only conditions which are not connected by the same letter are significantly different.

Menu, Motion Level				**Least Sq Mean**
Seven, Fast	A			1,242.96
Seven , Stand	A			1,218.41
Five , Fast	A			1,171.58
Seven , Slow	A			1,151.91
Five , Slow		B		984.67
Five , Stand		B		983.18
Three , Slow		B	C	950.71
Three , Stand		B	C	924.93
Three , Fast			C	823.40

Table 8.2: Menu X motion level interaction effect on *Preparation Time*: Only conditions which are not connected by the same letter are significantly different

nificantly slower than walking slowly (LMS = 984.67 ms) (B) and standing (LMS = 983.18 ms) (B), using the same menu size five. Figure 8.7 shows results regarding *Preparation Time*, along with error-bars representing 95% standard error (SE).

Pulling force level had a significant main-effect on *Preparation Time*.

Furthermore, pulling force level had a significant effect on *Preparation Time* ($F_{2,954}$ = 201.44, p < .0001). Tukey HSD posthoc pairwise comparisons between different force levels were all significant, i.e., *Preparation Time* decreased with level. Significant difference between different levels is illustrated in Table 8.3. Each level is referred to by different letter (A, B, or C). *Preparation Time* for highest level, referred to by the letter (A), was significantly shorter (LMS = 889.26 ms) than the time for both the middle level, referred to by

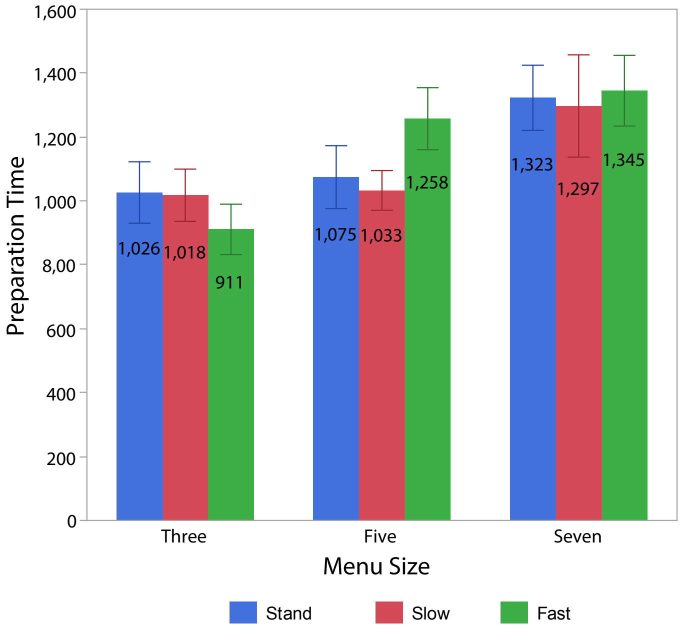


Figure 8.7: Menu size X motion level Interaction effect on *Preparation Time* (Error bars denote 95% confidence intervals (CIs).

letter (B), and the low level, referred to by letter (C). In addition, the table shows that the middle level also had a significantly shorter *Preparation Time* (LMS = 1,032.43 ms) than the time needed for the lowest level (1,227.87 ms). Figure 8.8 shows a a bar chart which shows both the mean *Preparation Time* and *Selection Time* along with 95% confidence intervals (CIs).

Pulling force level had a significant main-effect on *Selection Time.*

Pulling force level had a significant effect on the *Selection Time* ($F_{2,954}$ = 49.75, p < .0001). Tukey HSD posthoc pairwise comparisons between different force levels were all significant. Significant difference between different levels is illustrated in Table 8.3. *Preparation Time* for lowest level, referred to by the letter (A), was significantly shorter (LMS = 1.126.36 ms) than the time for both the high level, referred to by letter B, and the middle level, referred to by letter C. The table also shows that the high level also had a significantly shorter *Preparation Time* (LMS = 1,419.43 ms) than the time needed for the middle level (LMS = 1,688.24 ms).

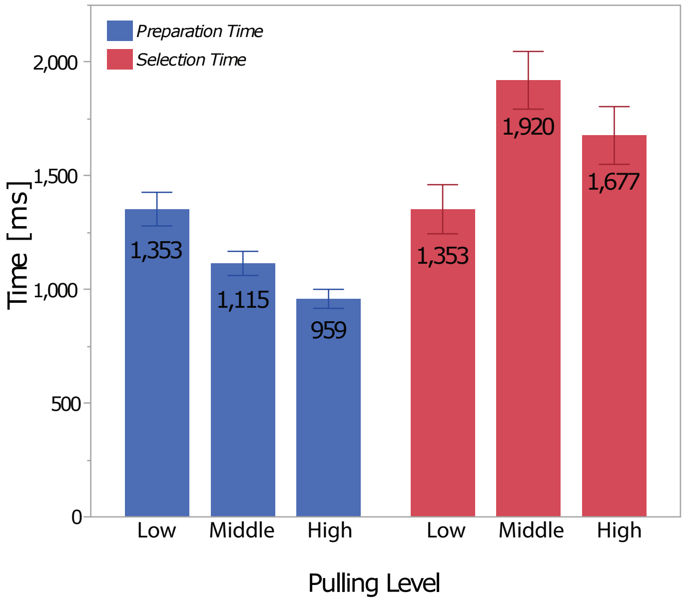


Figure 8.8: Means of *Preparation Time* and *Selection Time* [ms] according to pulling force level (Error bars denote 95% confidence intervals (CIs).

	Pulling Level				Least Sq Mean
Preparation Time	**Low**	A			1,227.87
	Middle		B		1,032.43
	High			C	889.26
Selection Time	**Middle**	A			1,688.24
	High		B		1,419.21
	Low			C	1,126.36

Table 8.3: Significant difference with the resulting *Preparation Time*, and *Selection Time* depending on pulling force level. Only conditions which are not connected by the same letter are significantly different.

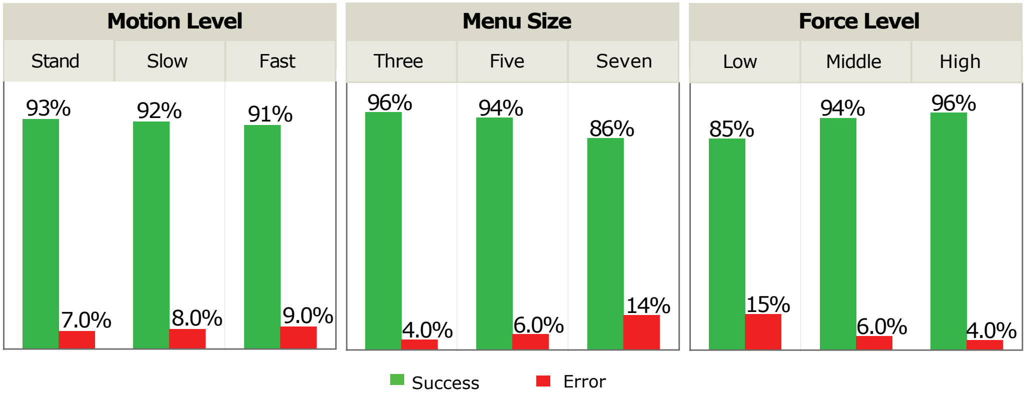


Figure 8.9: Results for *Target Accuracy* [%]. Both success and error rates are stated.

8.8.2 Target Accuracy

The following section will present the results of independent variables on *Target Accuracy*.

No significant effect on making errors ($X^2(2) = .502$, n.s.). In other words, in-motion interaction did not affect *Target Accuracy* rate. Figure 8.9 shows high *Target Accuracy* rates across the three different motion levels with 93% *Target Accuracy* rate while the user is standing still, 92% while walking slowly at 2.5 km per hour, and 91% while walking fast at 5km per hour.

Motion level had no effect on the *Target Accuracy*.

Menu size had a significant effect on *Target Accuracy* ($Q^2 = 28.441$, $p < .0001$). Posthoc pairwise McNemar tests revealed that *Target Accuracy* for menu 7 (85.0%) was significantly lower compared to menu 5 (93.0%) and 3 (95.0%) ($p < .0001$, each), i.e., users tend to make much more wrong selections with menu size of seven menu items (Figure 8.9).

Menu size had significant effect on *Target Accuracy*.

Pulling level had a significant effect on *Target Accuracy* ($Q^2 = 25.78$, $p < .0001$). Posthoc pairwise McNemar tests revealed that *Target Accuracy* for low level (85.0%) was significantly lower compared to middle level (93.0%) and high level (95.0%) ($p < .0001$, each), i.e., users tend to make much more false selections with the low force level (Figure 8.9).

Pulling force level had significant effect on *Target Accuracy*.

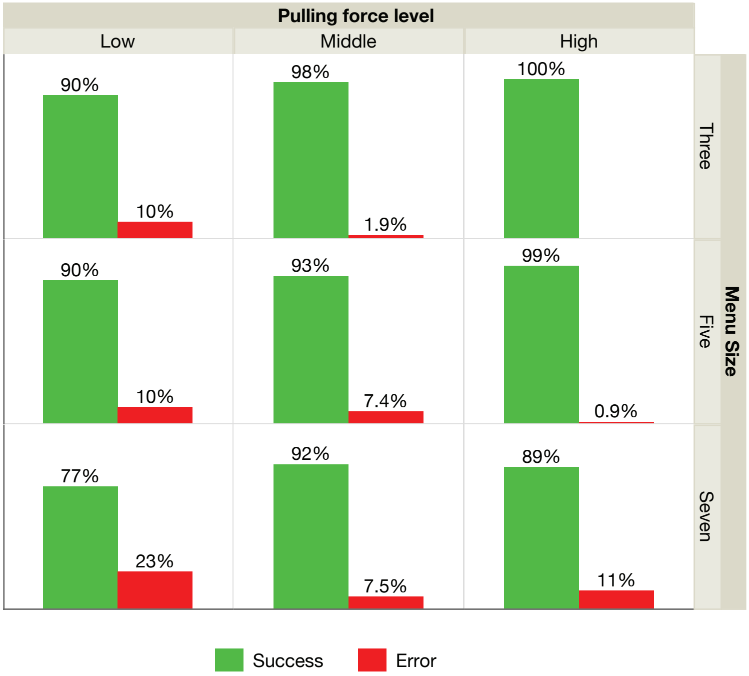


Figure 8.10: *Target Accuracy* of force levels in different menu sizes: *Target Accuracy* is almost perfect in menu size 3 using both the high force level (100%) and the middle force level (98%). However, the low force level has a much less *Target Accuracy* (90%).

Figure 8.10 provides a deeper look, although not significant, of where users tended to make the most errors. The figure illustrates success and error rates of the tasks with the respected force levels and menu sizes. Please note how the *Target Accuracy* rate for low force level in the three menu items is only 90% whereas it is 98% for the middle level and 100% for the high level.

8.8.3 Questionnaire

Participants showed satisfaction using PullBand.

Figure shows 8.11 shows all the 12 users were satisfied by using the band for selecting targets while both standing and in-motion. Users found that the concept works well for in-

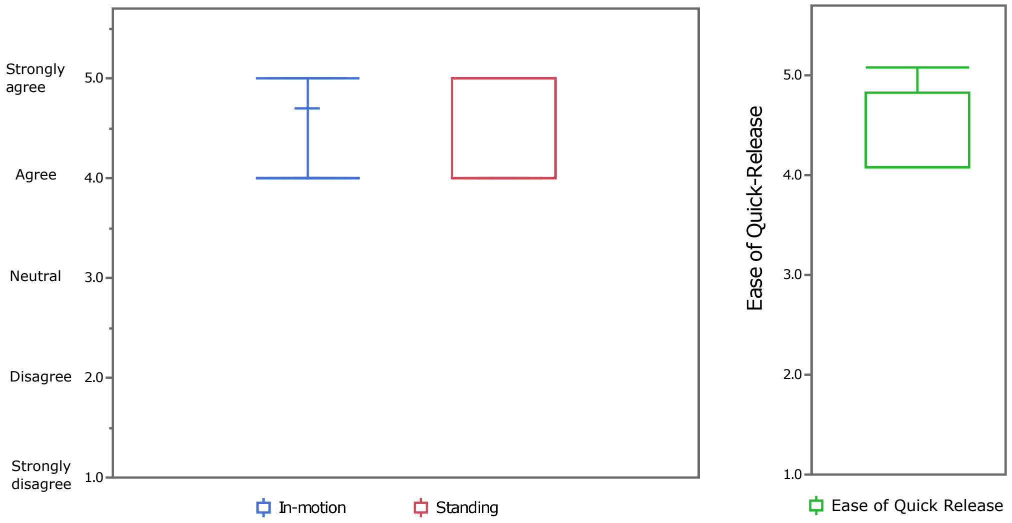


Figure 8.11: Questionnaire results on a 5 Likert scale where 1 (strongly disagree) and 5 (Strongly Agree). User were satisfied with PullBand as a selection technique for eyes-free and in-motion interaction.

motion interaction and does not cause any problems. Users also agreed or strongly agreed that Quick Release is easy and usable.

8.8.4 Discussion

There is no record that motion level has any significant effect on any of the *dependent variables*. Users needed similar amount of time to plan and start their interaction. In addition, users took a similar amount of times to actually select the targets. *Target Accuracy* was also similar (91%, 92%, 93%) respectively for (standing, walking slowly, fast walking). These findings support our claim that the user can use this technique for in-motion interaction.

Motion level did not have any significant effect on the user's performance.

Not surprisingly, both *Preparation Time* and *Selection Time* were shorter with smaller menu sizes. These results were expected as we anticipated the user to spend more time planning the interaction with larger menus than with smaller ones. In addition, the larger the menu the smaller the range of each item making it harder to select. In general,

As expected, both *Preparation Time* and *Selection Time* were shorter with smaller menu sizes.

users needed the shortest *Preparation Time* with the 3 menu items (M = 985 ms) followed by 5 menu item (M = 1,122 ms) and 7 menu items (M = 1322 ms). Similarly, *Selection Time* was the shortest (M = 1,113 ms) with 3 menu items, followed by (M = 1494 ms) with the 5 menu item. The 7 menu items needed the longest *Selection Time* (M = 2,434 ms), which is almost more than twice the time needed for the three menu items to be selected. Furthermore, *Target Accuracy* was significantly higher with smaller menu sizes (96%, 94%, 86%) for (three, five, and seven) menu sizes.

Target accuracy was higher with smaller menu sizes.

It should be emphasized that we recorded the interaction effect on *Preparation Time* in the five menu items while walking fast. Users needed more time in this condition (1,258 ms) to plan their interaction than when they were standing (1,075 ms) or walking slow (1,033 ms) with the same menu size (Figure 8.7). This result could be explained by the fact that the cognitive load that is occupied for the 5 menu item resulted in delay. However, we have no explanation why this interaction effect did not appear in the seven menu items while walking fast. A possible explanation could go back to the fact that the seven menu items needed more preparation times in all motion levels anyway. However, this is just an assumption and further evaluation, with larger number of participants, might be needed to investigate these results.

With 5 menu items, users needed more time to plan their interaction while they were walking fast.

Moreover, as expected, the middle pulling force level needed longer *Selection Time* (M = 1,920 ms) than the low and high levels. This is understandable as finding middle items takes more time than finding items at the end of ranges. The highest force level had, in general, higher *Selection Time* (M = 1,677 ms) than the first level (M = 1,353 ms) as it is more far to be reached and selected.

User needs the longest times to select the targets in the middle of the menu.

As we mentioned in Section 8.5.1 in the seven menu size, the high and low force levels were mapped into middle items (2nd, 6th). Results, although not significant, show that they had much higher *Selection Time* and error rate than what they had in the menu sizes five and three (Figure 8.10 and 8.12).

Regarding *Preparation Time*, the highest pulling level needed the shortest *Preparation Time* (M = 959 ms). This is not surprising as the user has only to think of applying maximum pulling level to land on the highest pulling level and this requires a less cognitive load to think and plan. The middle level came second (M = 1,115 ms). But, what is the most interesting is that the user needed the longest amount of *Preparation Time* (M = 1,353 ms) for the low level. This finding comes along with the *Target Accuracy* rates that showed a significantly lower *Target Accuracy* rate (85%) for the low pulling level than the other middle (94%) and high (96%) pulling levels. These findings combined indicate interaction problem with the first pulling level. Moreover, as we mentioned before, *Selection Time* for the low level was, as anticipated, the shortest due to the fact that it is the closest to be reached and selected. Taking a relatively long time to plan the interaction, but then having a quick selection process that ends up with a wrong confirmation indicates that there is a problem with the confirmation at this level.

Interestingly, the first pulling level had the longest *Preparation Time* and the lowest *Target Accuracy* rates.

To investigate that, we look into the continuous data log that we collected from the user study. We specifically look at the trials of the first pulling level with false selection (Figure 8.13). We found that users tended to slightly overshoot the target. In addition, the overshooting is correlated with a slight peek just before quick releasing. We assume the reason is related to the fact that the user's finger is relatively close to the user's wrist, when selecting the first pulling level, making it harder to be released. We expect that users adopted a strategy where they applied a slightly extra puling once they were snapping their finger out so it does not get stuck. It is important to remind the reader, that we added a threshold before the first level in order to make it easier to apply quick Release after the threshold range. A possible solution is to make this threshold bigger until the limit where the user is comfortable to apply Quick Release.

Users tend to slightly overshoot the first pulling level.

8.8.5 Conclusion

All in all, this user study indicates that our technique performs as good as in stationary mode and in-motion with a

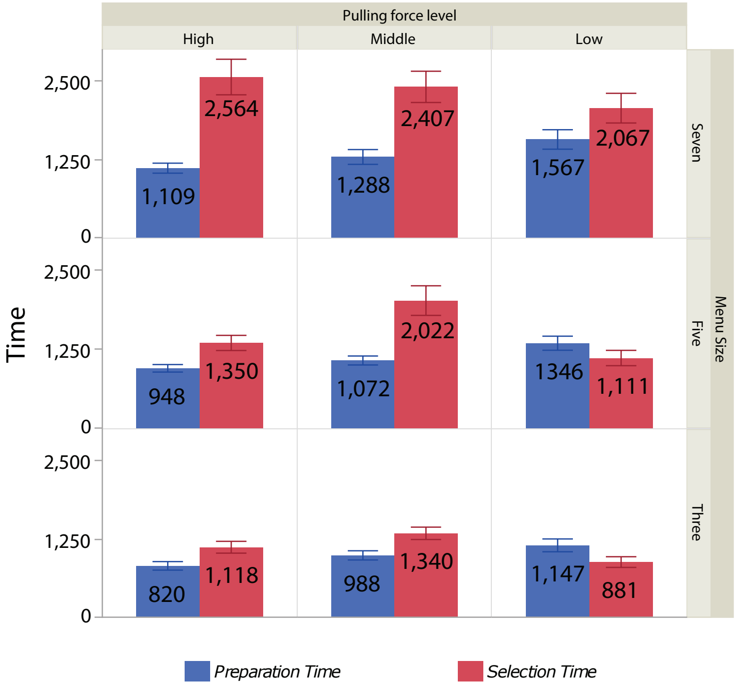


Figure 8.12: *Selection time* of force levels in different menu sizes: the high and low levels in menu size (seven) need longer *Selection Time* than what they need in menu size 3 and 5.

PullBand, supports selection tasks on three menu items with 96% *target accuracy*.

speed up to 5 km per hour. These results support our claim about introducing an eyes-free in-motion interaction technique. The user can easily perform discrete selection tasks, eyes-free with 96% success rate. Although the difference was significant compared to the five menu items, 94% success rate, it was still very small. The above-stated findings also revealed that the highest force level was the easiest to select.

Users had the highest error rates of 15% confirming his selection on the first force level.

The technique still faces a problem selecting the first pulling level. This was seen through the significantly longer *Preparation Time* and higher error rate. Possible reason could be related to difficulty of performing quick releasing at low level. Further investigation is required.

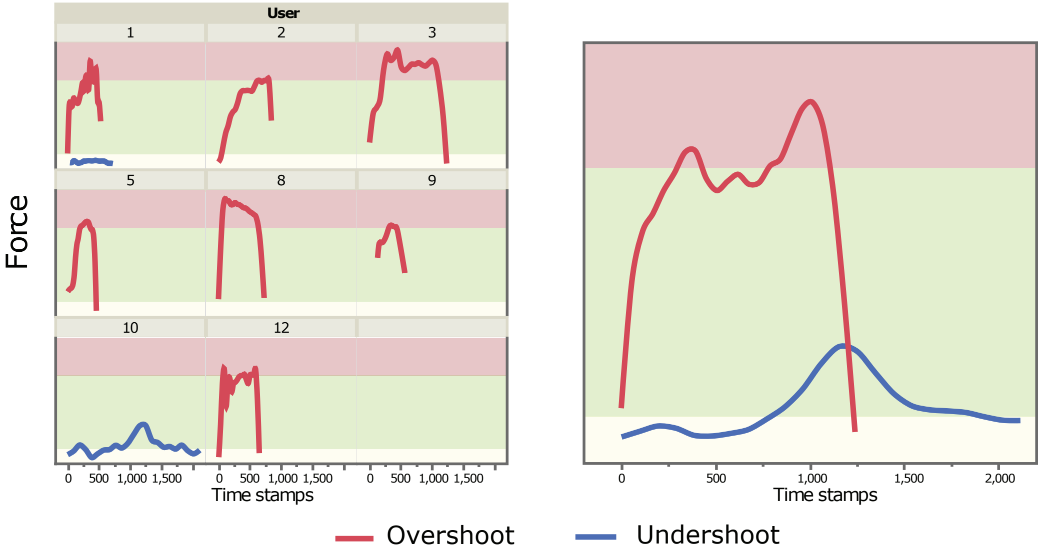


Figure 8.13: Continuous data log of the trials, taken from several users while selecting the first pulling force level and ending up with false selection. Users tend to overshoot the first level.

To summarize, this chapter evaluated the technique in terms of user performance using different motion levels, menu sizes, and force levels. We also identified the conditions in which the technique performs the best. In addition, we analyzed the existing problems with the interaction technique, especially in the mechanism of supporting Quick Release. Next chapter refers to the summary of the work, limitations, and suggested future work.

Chapter 9

Summary and Future Work

9.1 Summary

This work aims to utilize the act of deformation on the watch band as an alternative input modality.

The watch screen is the main interaction possibility when it comes to smartwatch interaction. However, it suffers from the fat finger and the visual occlusion problems related to its small size factor. Depending on the watch screen also forces another major flaw when it comes to in-motion and eyes-free interaction. The majority of the related work has not considered in-motion and eyes-free interaction scenarios or only designed techniques without actually evaluating them in such conditions. In this manner, previous work has pointed out that utilizing the act of deformation on the watch band could provide a solution to the abovementioned problems. The main objective of this thesis was to design an interaction technique, which provides tactile feedback by utilizing the act of deformation on the band, supporting in-motion and eyes-free interaction.

We started our iterative design process by identifying several ideas from the related work that provided an affordance for bending, twisting, and pulling. We also explored various deformational sensing techniques to come up with additional design ideas.

We explored several deformational interactions and chose the most preferred one through a preliminary user study.

During our low-fidelity prototyping, we designed several mock-ups that give an affordance for bending, twisting, and pulling. We conducted a preliminary user study to explore the user's preference and the satisfaction levels for these techniques. We were also interested in the most preferred location on the band to perform these interactions. Results suggested that PullBand is the most preferred prototype. In addition, results regarding this specific technique showed that users preferred to perform pulling at the bottom area of the band and only in one direction. Users comments also helped us to identify problems regarding the loose cord design of this technique. Users reported that it could get easily stuck while interacting with every day objects. Comments also refereed to a difficulty of finding the cord eyes-free as it lacked any reference point. Therefore, we redesigned this prototype to utilize the main watch band instead of attaching it as an extra cord. This new design also offered a direct contact with the user's skin which acts not only as a reference point useful for eyes-free interaction but also provides more feel of control on the wearing hand.

PullBand provides accurate measurements and supports quick release as a selection technique

To implement the software and the hardware of PullBand, we placed a force sensitive sensor on the back side of the watch face, allowing us to detect pulling when it is performed on the bottom area of the band. The force sensor offered the ability to utilize pulling as a menu selection technique, which stretch sensing techniques are not capable of. Using the force sensor, we accurately measure the applied pulling force and map it to discrete menu items. Using the force sensor also allowed us to utilize quick releasing the band as a confirmation mechanism, also which the stretch sensors fail to offer. In order to provide robust measures with zero noise, we designed a multi-layered sensor casing which is placed on the back of the watch. Furthermore, we connected the sensors to Apple Watch and iPhone using BLE Redbear interface.

We continued to evaluate the appropriateness of our high-fidelity prototype in terms of user's performance. For that, we implemented an automated user study to measure *Preparation Time*, *Selection Time*, and *Target Accuracy*. We evaluated our technique against audio menu selection tasks using different menu sizes, pulling force levels, and motion levels. Findings revealed that users had similar performance in all motion levels (0-5 km per hour). In addition, it is possible to state that three menu items performed the best in all different motion levels with *Target Accuracy* (96%) and *Selection Time* (M = 1,113 ms). Five menu items had less *Accuracy* results (94%) and longer *Selection Time* (M = 1,494 ms) but with a slight difference from the three menu items. On the other hand, the seven menu items showed a serious flaw in the user's performance. Furthermore, in terms of the applied pulling force, users had the best performance while applying a maximum amount of pulling force. On the other hand, it took the longest *Selection Time* (M = 1,920 ms) to accurately select items applying middle amount of pulling force. Interestingly, users had the lowest *target accuracy* (85%) selecting the first menu items since it was relatively difficult to apply quick release when the finger is too close to the user's skin.

Motion level had no effect on user's performance using PullBand

Having gone through the summary, it is worth mentioning that it would be also interesting to develop new applications that are accessible while in-motion using PullBand. As an example application of our technique, we implemented a music application where we used PullBand to perform micro interactions such as play/pause, play next, and play previous while in-motion and eyes-free. In this implementation, we used the menu size of three menu items and assigned a larger pulling range for the middle item hoping for a quicker selection. We also raised the threshold of the first menu item to make it easier for the user to apply quick releasing using the low pulling level. Moreover, to simplify the selection process we used the design guidelines by Oakley and Park [2007] and assigned the most accessed command "play/pause to the easiest level to select the last menu item. In addition, we placed less accessible commands, namely, next song and previous song under more difficult to select menu items, namely, the first and the second menu items. By that we have also mapped the related commands (next, previous) next to each other so it becomes easier for the user to learn and memories as suggested by Oakley and Park [2007].

We implemented a music app to show how PullBand could be used while in-motion and eyes-free.

9.2 Limitations and Future Work

The suggested adjustments could be evaluated in a further user study.

We presented in our music application example, how we used the results from the PullBand evaluation user study to make it quicker to select when applying middle amount of force, and easier to confirm at first menu item. However, a follow-up user study should be conducted to explore whether these adjustments can really make a difference.

In addition, in our user study, we evaluated PullBand in different motion levels (up to 5 km per hour) using a treadmill. However, a further evaluation could be performed using a faster speed and more challenging environment, where obstacles could be placed, leaving less amount of cognitive load available for the interaction.

Further work could evaluate providing other types of audio and haptic feedback.

Furthermore, we presented our menu selection technique using an ordered domain, which allowed the user to estimate the target's position beforehand. It would be also interesting to evaluate the usage of PullBand using an unordered audio list where the user has to scroll through the items, one by one until he finds the target he is looking for. Future work could also explore applying feedback that could be scaled depending on experience. If a new user interacts with the menu, he will hear voice messages. However, for more experienced users only haptic feedback combined with short patterns of natural sounds could be provided Oakley and Park [2007]. It would be interesting to check the validity of such an approach in a user study. User could be provided with an auditory menu which will help him to learn how the system works. Once the user learns the system, the auditory feedback is aborted and only haptic feedback and short natural sounds are provided.

PullBand could be evaluated for on-screen target selection.

Moreover, since the main focus of our evaluation was to support in-motion and eyes-free interaction, we only evaluated our technique against an auditory menu. Although we provided a visual exploratory menu, for the user to understand the technique, we have not evaluated the user's performance using it. Future work could evaluate using our technique for on-screen target selection.

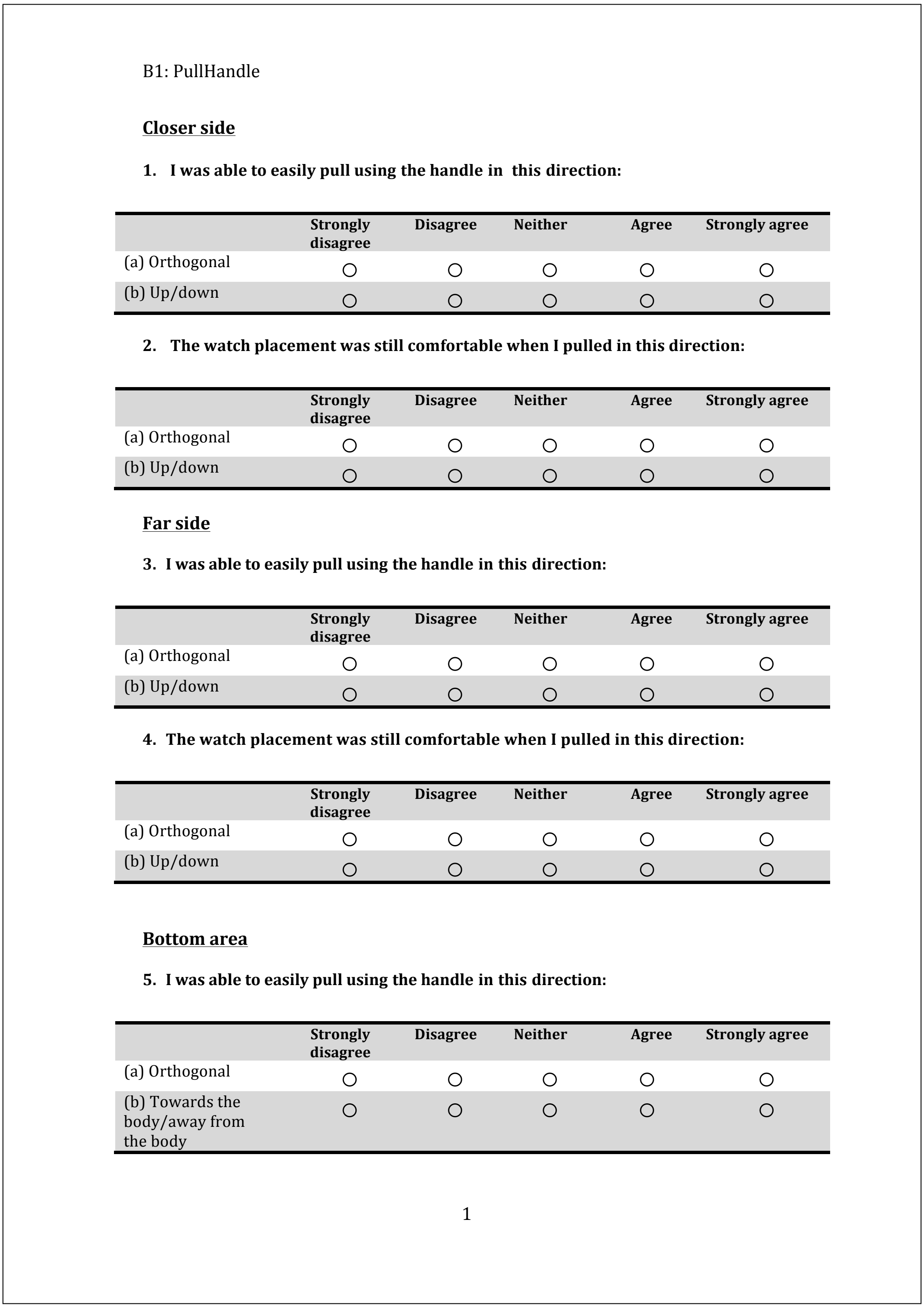
Last but not least, we would like to mention that our set-up implementation, using the force sensor on the back side of the watch face, allows us to detect force not only when the user pulls the band but also when the user pushes the watch face towards his wrist's skin. It could be also interesting, for future work, to evaluate pushing the watch face as a selection technique using our sensing set-up. Please note that, in our user study, users interacted with watch only by pulling the band and no pushing force was applied on the watch face. However, in future work, a textile fabricated force sensor could be included on the band to ensure detecting force only when pulling.

Our set-up could be also utilized as a selection technique using applied force on the watch face.

Appendix A

Preliminary User Study Questionnaire

The following questionnaire is used in the preliminary user study, participants are asked to state their *Satisfaction* levels, for each of the low fidelity prototypes, on a five-point Likert scale, i.e., from totally disagree to totally agree. This questionnaire is divided into four blocks, each represents one of the four prototypes: PullHandle, PullBand, FlexibleStraps-up, and FlexibleStraps-down. Participants are asked to fill each questionnaire block after interacting with the respected prototype. In addition, after all tasks are finished, one final sheet is handed to the users. In this sheet, user's are asked to state their *Preference* regarding the different prototypes through rankings from the most preferred technique, i.e., 1, to least preferred one, i.e., 4. It should be mentioned that participants were free to assign the same score to multiple techniques. Please not that comments are also collected regarding the prototype preference.



B1: PullHandle

Closer side

1. **I was able to easily pull using the handle in this direction:**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

2. **The watch placement was still comfortable when I pulled in this direction:**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

Far side

3. **I was able to easily pull using the handle in this direction:**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

4. **The watch placement was still comfortable when I pulled in this direction:**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

Bottom area

5. **I was able to easily pull using the handle in this direction:**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Towards the body/away from the body	○	○	○	○	○

B1: PullHandle

6. The watch placement was still comfortable when I pulled in this direction:

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Towards the body/away from the body	○	○	○	○	○

General

7. I felt confident to apply minimum pulling force

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

8. I felt confident to apply maximum pulling force

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

9. I was able to easily pull and quickly release the handle

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

10. I was able to easily pull and hold the handle

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

11. I was able to easily pull repeatedly

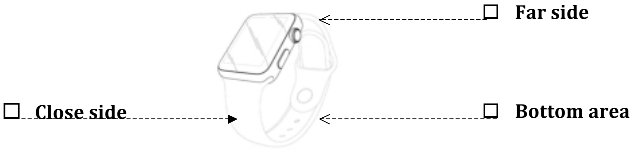
	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

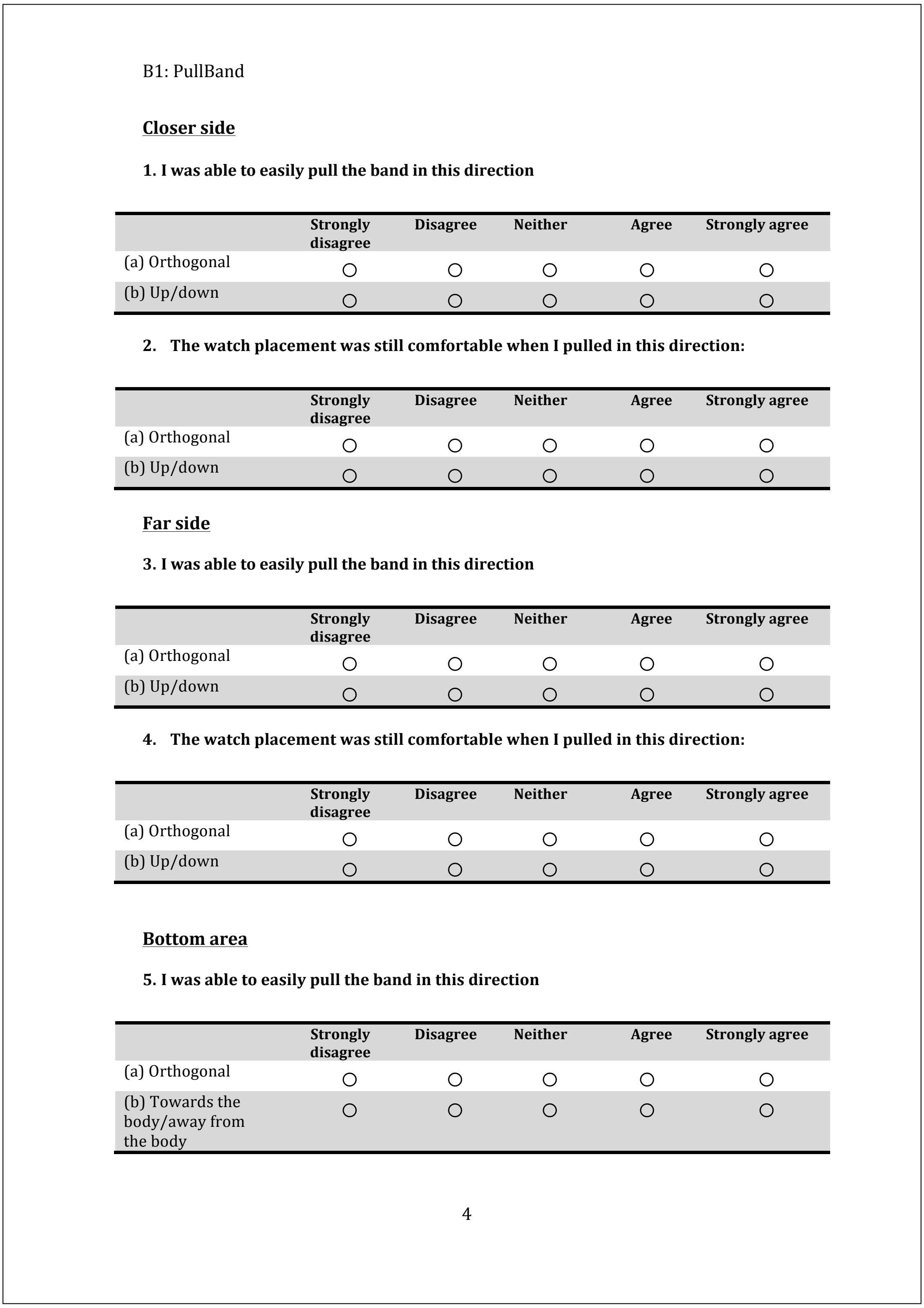
B1: PullHandle

12. I was able to easily pull without looking at the watch

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

13. In my opinion, it is convenient to interact with PullHandle on the following areas: (Check all that apply)





B1: PullBand

Closer side

1. I was able to easily pull the band in this direction

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

2. The watch placement was still comfortable when I pulled in this direction:

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

Far side

3. I was able to easily pull the band in this direction

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

4. The watch placement was still comfortable when I pulled in this direction:

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
(a) Orthogonal	○	○	○	○	○
(b) Up/down	○	○	○	○	○

Bottom area

5. I was able to easily pull the band in this direction

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
(a) Orthogonal	○	○	○	○	○
(b) Towards the body/away from the body	○	○	○	○	○

B1: PullBand

6. The watch placement was still comfortable when I pulled in this direction:

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Orthogonal	○	○	○	○	○
(b) Towards the body/away from the body	○	○	○	○	○

General

7. I felt confident to apply minimum pulling force

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

8. I felt confident to apply maximum pulling force

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

9. I was able to easily pull and quickly release the string

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

10. I was able to easily pull and hold

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

11. I was able to easily pull repeatedly

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

12. I was able to easily pull without looking at the watch

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

B2: FlexibleStraps- up

Closer side

1. I was able to easily bend the flexible strap

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

2. I was able to easily twist the flexible strap in this direction

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Left	○	○	○	○	○
(b) Right	○	○	○	○	○

Far side

3. I was able to easily bend the flexible strap

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

4. I was able to easily twist the flexible strap in this direction

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Left	○	○	○	○	○
(b) Right	○	○	○	○	○

Bottom area

5. I was able to easily bend the flexible strap

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

6. I was able to easily twist the flexible strap in this direction

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Left	○	○	○	○	○
(b) Right	○	○	○	○	○

B2: FlexibleStraps- up

General

7. **I felt confident bending the flexible straps far away until maximum flexibility was reached**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

1. **I felt confident snapping the flexible straps**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

2. **I was able to easily snap the flexible straps repeatedly**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

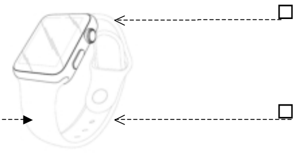
3. **I was able to easily bend and hold the flexible straps**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

4. **I was able to easily bend/twist without looking at the watch**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

5. **In my opinion, it is convenient to have flexible straps on all of the following areas: (Check all that apply)**



☐ **Far side**

☐ **Close side**

☐ **Bottom area**

B2: FlexibleStraps- down

Closer side

1. **I was able to easily bend the flexible strap**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

2. **I was able to easily twist the flexible strap in this direction**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Left	○	○	○	○	○
(b) Right	○	○	○	○	○

Far side

3. **I was able to easily bend the flexible strap**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

4. **I was able to easily twist the flexible strap in this direction**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Left	○	○	○	○	○
(b) Right	○	○	○	○	○

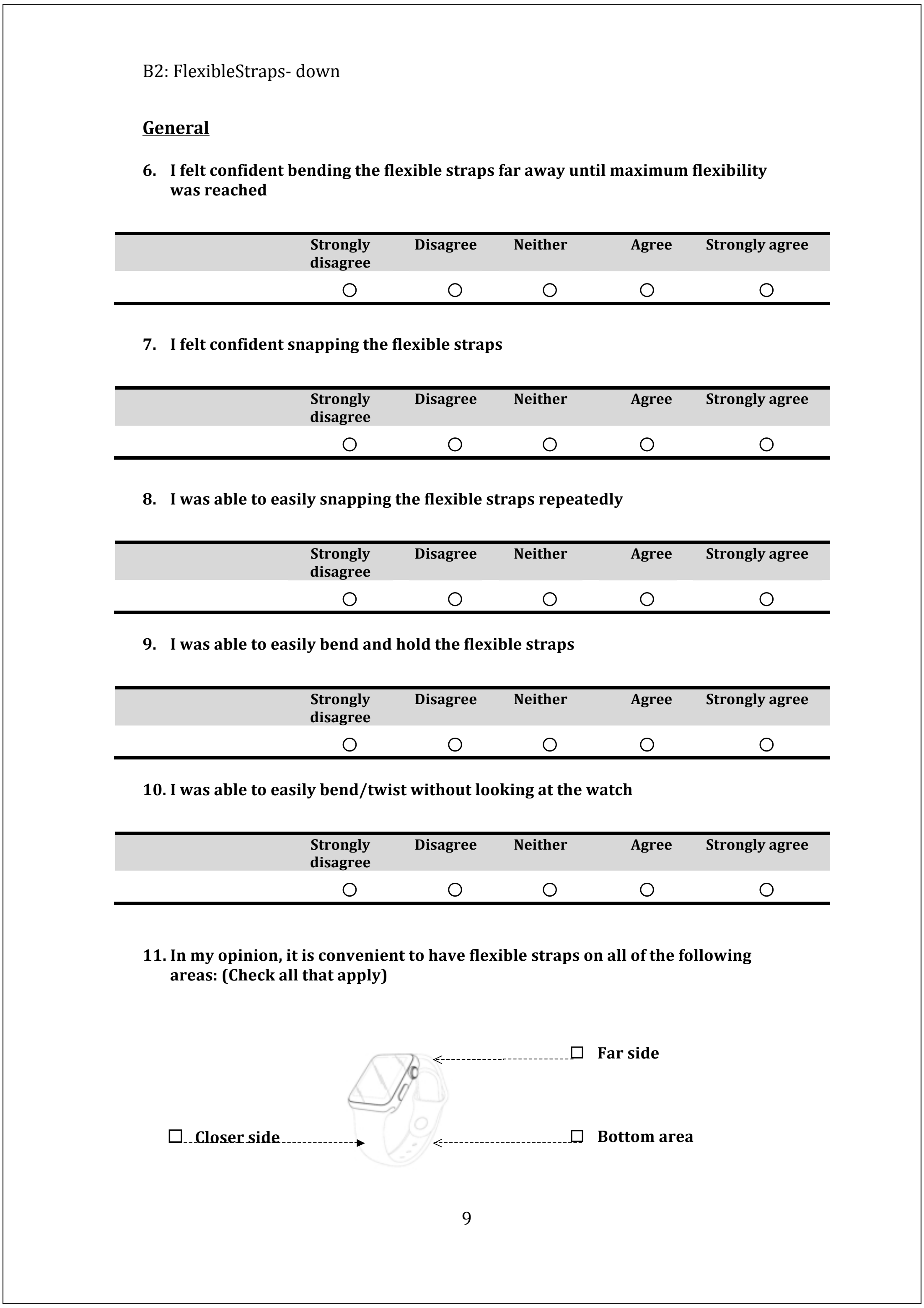
Bottom area

5. **I was able to easily bend the flexible strap**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

6. **I was able to easily twist the flexible strap in this direction**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Left	○	○	○	○	○
(b) Right	○	○	○	○	○



B2: FlexibleStraps- down

General

6. **I felt confident bending the flexible straps far away until maximum flexibility was reached**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

7. **I felt confident snapping the flexible straps**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

8. **I was able to easily snapping the flexible straps repeatedly**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

9. **I was able to easily bend and hold the flexible straps**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

10. **I was able to easily bend/twist without looking at the watch**

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
	○	○	○	○	○

11. **In my opinion, it is convenient to have flexible straps on all of the following areas: (Check all that apply)**

☐ **Far side**

☐ **Closer side**

☐ **Bottom area**

Final

8. I was able to easily reach this area:

	Strongly disagree	Disagree	Neither	Agree	Strongly agree
(a) Closer side	○	○	○	○	○
(b) Far side	○	○	○	○	○
(c) Bottom area	○	○	○	○	○

9. I rank the previous techniques from 1 to 3 as follow (1 most preferred, 4 least preferred)

	1	2	3	4	No opinion
(a) Pulling PullHandle	○	○	○	○	○
(b) Pulling PullBand	○	○	○	○	○
(c) Bending	○	○	○	○	○
(d) Twisting	○	○	○	○	○

10. Please fill in the gap If you have any comments

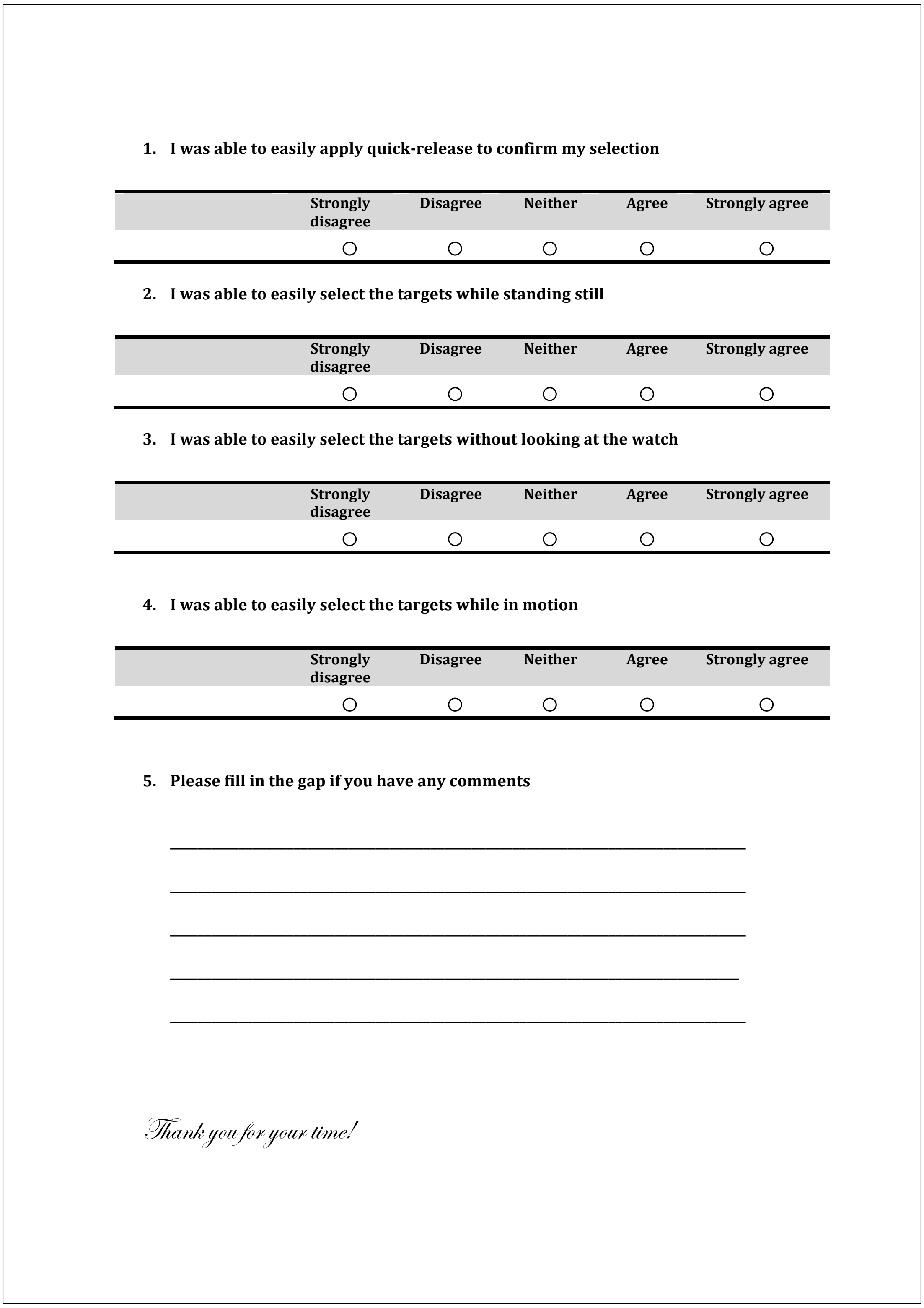
____
____
____
____
____
____
____
____
____

Thank you for your time!

Appendix B

PullBand Evaluation Questionnaire

The evaluation of PullBand, our high fidelity prototype, is focused on collecting the user's quantitative performance in terms of *Preparation Time*, *Selection Time* and *Target Accuracy*. In addition to that, the following short questionnaire is handed after the study to collect general qualitative data. We mainly ask the users about their *Satisfaction* levels regarding PullBand, our high fidelity prototype. Users are also asked to state their comments and suggestions regarding problems or further improvements.



1. **I was able to easily apply quick-release to confirm my selection**

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
	○	○	○	○	○

2. **I was able to easily select the targets while standing still**

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
	○	○	○	○	○

3. **I was able to easily select the targets without looking at the watch**

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
	○	○	○	○	○

4. **I was able to easily select the targets while in motion**

	Strongly disagree	**Disagree**	**Neither**	**Agree**	**Strongly agree**
	○	○	○	○	○

5. **Please fill in the gap if you have any comments**

____

____

____

____

____

Thank you for your time!

Bibliography

Adafruit. https://www.adafruit.com/product/519. Accessed: 2018-12-02.

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*, pages 251–254. ACM, 2015.

Apple. Apple Watch. https://www.apple.com/lae/watch/, 2018. Accessed: 2018-12-02.

Apple. Core Bluetooth. https://developer.apple.com/documentation/corebluetooth, 11 2018. Accessed: 2018-12-02.

Karen Aspelin. Establishing pedestrian walking speeds. *Portland State University*, pages 5–25, 2005.

Patrick Baudisch and Gerry Chu. Back-of-device interaction allows creating very small touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1923–1932. ACM, 2009.

Xiaojun Bi, Yang Li, and Shumin Zhai. FFitts law: modeling finger touch with fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1363–1372. ACM, 2013.

Stephen Brewster, Joanna Lumsden, Marek Bell, Malcolm Hall, and Stuart Tasker. Multimodal'eyes-free'interaction techniques for wearable devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 473–480. ACM, 2003.

Bill Buxton. Bill Buxton collection. `https://www.microsoft.com/buxtoncollection/type.aspx?t=Watch`. Accessed: 2018-11-17.

SK Card and Th P Moran. Newell (1983) The Psychology of Human-Computer Interaction. *The psychology of human-computer interaction. CRC*, page 488.

Victor Cheung, Alex Keith Eady, and Audrey Girouard. Exploring Eyes-free Interaction with Wrist-Worn Deformable Materials. In *Proceedings of the Tenth International Conference on Tangible, Embedded, and Embodied Interaction*, pages 521–528. ACM, 2017.

William G Cochran and Gertrude M Cox. Experimental designs. 1950.

Christian Corsten, Simon Voelker, and Jan Borchers. Release, Don't Wait!: Reliable Force Input Confirmation with Quick Release. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, pages 246–251. ACM, 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*, pages 2431–2438. ACM, 2016.

Alan Dix. Human-computer interaction. In *Encyclopedia of database systems*, pages 1327–1331. Springer, 2009.

EPSON. TV Watch. `http://www.guenthoer.de/doku/werbung-seikotvwatch.pdf`, 11 1982. Accessed: 2018-11-17.

Google. wear OS. `https://www.android.com/wear/`. Accessed: 2018-12-02.

Anhong Guo and Tim Paek. Exploring tilt for no-touch, wrist-only interactions on smartwatches. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 17–28. ACM, 2016.

Chris Harrison and Scott E Hudson. Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 121–124. ACM, 2009.

Bertrand Hochet and Antonio J Acosta. *Integrated Circuit Design. Power and Timing Modeling, Optimization and Simulation: 12th International Workshop, PATMOS 2002, Seville, Spain, September 11-13, 2002*, volume 12. Springer Science & Business Media, 2002.

Jungsoo Kim, Jiasheng He, Kent Lyons, and Thad Starner. The gesture watch: A wireless contact-free gesture based wrist interface. In *Wearable Computers, 2007 11th IEEE International Symposium on*, pages 15–22. IEEE, 2007.

Sven Kratz and Michael Rohs. Hoverflow: exploring around-device interaction with IR distance sensors. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, page 42. ACM, 2009.

Gierad Laput, Robert Xiao, Xiang'Anthony' Chen, Scott E Hudson, and Chris Harrison. Skin buttons: cheap, small, low-powered and clickable fixed-icon laser projectors. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pages 389–394. ACM, 2014.

Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research methods in human-computer interaction*. Morgan Kaufmann, 2017.

Clayton Lewis. Using the'thinking-aloud'method in cognitive interface design. *Research Report RC9265, IBM TJ Watson Research Center*, 1982.

Kent Lyons, David Nguyen, Daniel Ashbrook, and Sean White. Facet: a multi-segment wrist worn system. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 123–130. ACM, 2012.

Joe Marshall and Paul Tennent. Mobile interaction does not exist. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 2069–2078. ACM, 2013.

Joe Marshall, Alexandru Dancu, and Florian Floyd Mueller. Interaction in motion: designing truly mobile interaction.

In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, pages 215–228. ACM, 2016.

Chulhong Min, Seungwoo Kang, Chungkuk Yoo, Jeehoon Cha, Sangwon Choi, Younghan Oh, and Junehwa Song. Exploring current practices for battery use and management of smartwatches. In *Proceedings of the 2015 ACM international symposium on wearable computers*, pages 11–18. ACM, 2015.

Terhi Mustonen, Maria Olkkonen, and Jukka Hakkinen. Examining mobile phone text legibility while walking. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 1243–1246. ACM, 2004.

Chandrasekhar Narayanaswami. Linux watch. `https://researcher.watson.ibm.com/researcher/view_group_subpage.php?id=6104`. Accessed: 2018-12-02.

Ian Oakley and Jun-Seok Park. Designing eyes-free interaction. In *International Workshop on Haptic and Audio Interaction Design*, pages 121–132. Springer, 2007.

Masa Ogata and Michita Imai. SkinWatch: skin gesture interaction for smart watch. In *Proceedings of the 6th Augmented Human International Conference*, pages 21–24. ACM, 2015.

Stephen Oney, Chris Harrison, Amy Ogan, and Jason Wiese. ZoomBoard: a diminutive qwerty soft keyboard using iterative zooming for ultra-small devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2799–2802. ACM, 2013.

Minna Pakanen, Ashley Colley, Jonna Häkkilä, Johan Kildal, and Vuokko Lantz. Squeezy bracelet: designing a wearable communication device for tactile interaction. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, pages 305–314. ACM, 2014.

Jerome Pasquero, Scott J Stobbe, and Noel Stonehouse. A haptic wristwatch for eyes-free interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3257–3266. ACM, 2011.

Simon T Perrault, Eric Lecolinet, James Eagan, and Yves Guiard. Watchit: simple gestures and eyes-free interaction for wristwatches and bracelets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1451–1460. ACM, 2013.

Mahfuz Rahman, Sean Gustafson, Pourang Irani, and Sriram Subramanian. Tilt techniques: investigating the dexterity of wrist-based input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1943–1952. ACM, 2009.

Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. Pressure widgets. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 487–494. ACM, 2004.

RedBear. BLE Nano v1. `https://redbear.cc/ble-nano.html`. Accessed: 2018-12-02.

Garyn G Roberts. *Dick Tracy and American culture: Morality and mythology, text and context*. McFarland, 1993.

Léa Saviot, Frederik Brudy, and Steven Houben. WRISTBAND. IO: expanding input and output spaces of a Smartwatch. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2025–2033. ACM, 2017.

Bastian Schildbach and Enrico Rukzio. Investigating selection and reading performance on a mobile phone while walking. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, pages 93–102. ACM, 2010.

Paden Shorey and Audrey Girouard. Bendtroller:: An Exploration of In-Game Action Mappings with a Deformable Game Controller. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 1447–1458. ACM, 2017.

The Statistics Portal Statista. Smartwatch unit sales worldwide from 2014 to 2018 (in millions). `www.statista.com/statistics/538237/global-smartwatch-unit-sales/`, 06 2017. Accessed: 2018-06-10.

Ke Sun, Yuntao Wang, Chun Yu, Yukang Yan, Hongyi Wen, and Yuanchun Shi. Float: One-Handed and Touch-Free Target Selection on Smartwatches. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 692–704. ACM, 2017.

Aku Visuri, Zhanna Sarsenbayeva, Niels van Berkel, Jorge Goncalves, Reza Rawassizadeh, Vassilis Kostakos, and Denzil Ferreira. Quantifying sources and types of smartwatch usage sessions. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3569–3581. ACM, 2017.

Anita Vogl, Patrick Parzer, Teo Babic, Joanne Leong, Alex Olwal, and Michael Haller. StretchEBand: Enabling Fabric-based Interactions through Rapid Fabrication of Textile Stretch Sensors. In *CHI*, pages 2617–2627, 2017.

Mark Wiser. The Computer for the 21st Century—Scientific American Special Issue on Communications. *Computers, and Networks*, pages 94–104, 1991.

Jacob O. Wobbrock. Practical Statistics for HCI. `https://depts.washington.edu/madlab/proj/ps4hci/`, 2011. Accessed: 2018-12-02.

Haijun Xia, Tovi Grossman, and George Fitzmaurice. Nanostylus: enhancing input on ultra-small displays with a finger-mounted stylus. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, pages 447–456. ACM, 2015.

Robert Xiao, Gierad Laput, and Chris Harrison. Expanding the input expressivity of smartwatches with mechanical pan, twist, tilt and click. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 193–196. ACM, 2014.

Hui-Shyong Yeo, Juyoung Lee, Andrea Bianchi, and Aaron Quigley. Sidetap & slingshot gestures on unmodified smartwatches. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 189–190. ACM, 2016a.

Hui-Shyong Yeo, Juyoung Lee, Andrea Bianchi, and Aaron Quigley. WatchMI: pressure touch, twist and pan gesture input on unmodified smartwatches. In *Proceedings* *of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 394–399. ACM, 2016b.

Cheng Zhang, AbdelKareem Bedri, Gabriel Reyes, Bailey Bercik, Omer T Inan, Thad E Starner, and Gregory D Abowd. Tapskin: Recognizing on-skin input for smartwatches. In *Proceedings of the 2016 ACM on Interactive Surfaces and Spaces*, pages 13–22. ACM, 2016a.

Yang Zhang, Junhan Zhou, Gierad Laput, and Chris Harrison. Skintrack: Using the body as an electrical waveguide for continuous finger tracking on the skin. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1491–1503. ACM, 2016b.

Index