UNIVERSITY OF COPENHAGEN
FACULTY OR DEPARTMENT

Master Thesis

Tudor-Stefan Dragan

Interacting Beyond The Smartwatch Display Using Motion Tracking Technologies

Investigating user performance and usability for off-screen mid-air pointing on smartwatches.

Supervisor: Sebastian Boring

Submitted on: 05/03/2017

Name of department:	**Department of Computer Science**
Author(s):	**Tudor-Stefan Dragan**
Title and subtitle:	**Interacting Beyond The Smartwatch Display Using Motion Tracking Technologies** Investigating user performance and usability for off-screen mid-air pointing on smartwatches.
Topic description:	The thesis investigates the effectiveness of off-screen mid-air pointing for smartwatches. It describes in detail the development of the mid-air tracker, the prototype used in the user study, as well as the results and guidelines for future development.
Supervisor:	Sebastian Boring
Submitted on:	05 March 2017
Grade:	[ ]

Abstract

Smartwatches have been gaining popularity in recent years because of the constant need for portability and accessing information on-the-go. Smartwatches have also evolved due to advancements in technology and the industry's improvements in the manufacturing process and industrial design. With hands truly free, smartwatch wearers can also read texts, see who is calling them, scan Twitter or Facebook feeds and of course, check the time. But current methods of interaction with such a small and limited screen real estate often creates frustrating issues with user input and occlusion of the displayed information, and multiple repetitive gestures to access desired user content.

The wearer is constantly panning, zooming and tapping to acquire information that lies beyond the edge of the screen. Interacting with graphical user interfaces is normally dependent on on-screen visual feedback; however, our experiences with interaction in the physical world suggest we could also take advantage of the region in the space around the device.

By transitioning into spatial target acquisition, the problem space changes. Precision, speed and ease of use are just some of the factors that come to mind when comparing to traditional methods of interaction.

This thesis will investigate *innovative motion tracking for interacting with small-screen devices*, in particular smartwatch displays, by extending the input space of a display into the space beyond the screen through the use of mid-air pointing. Most research is concentrated around large and smartphone displays, but not enough analysis has been made on smartwatch screens, due to the relatively new implementations and currently scarce user acquisition rates of smartwatches.

Preface

This thesis has been written at the Institute of Computer Science from the University of Copenhagen. This wouldn't have been made possible without the help and guidance of the people mentioned below.

Firstly, I would like to thank Sebastian for supervising me and providing me with weekly meetings that helped me carry out my work without important roadblocks. I would also like to thank my family and friends for helping and being so supportive. Last but not least, I would like to give very special thanks to my girlfriend, Sabrina, who has always believed in me and always helped me pass stressing times. I am forever grateful and I am hoping that I can do the same for her during her PhD endeavor.

Table of Contents

TABLE OF FIGURES

1 INTRODUCTION

The first pocket watch was invented by German locksmith Peter Henlein. Henlein's pocket clock picked up the name *watch* from sailors. Sailors replaced the hourglass they had used for timing the length of each of their watches (four-day duty shifts) with Henlein's pocket clock. Soon enough the clock itself was called a watch, and the name stuck forever since. The concept of the wristwatch goes back to the production of the very earliest watches in the 16th century. Elizabeth I of England received a wristwatch from Robert Dudley in 1571, described as an arm watch. From the beginning, wristwatches were almost exclusively worn by women, while men used pocket watches up until the early 20th century.[1]

Although the first commercially available wristwatch dates back 200 years ago [2], the digital watch came into existence in 1972 with the Pulsar, which was manufactured by the Hamilton Watch Company[3]. The Pulsar watch could store 24 digits, which made it the first commercial available watch with user programmable memory [4]. The first smartwatches were developed as a side effect of the introduction of personal computers in the 1980s. As technology improved and miniaturization became the prominent trend in manufacturing, the smartwatch took a more advanced and useful form. Today's smartwatches integrate CPUs comparable to a desktop's from 1990s[5] and incorporate numerous sensors like accelerometers, gyroscopes and altimeters for tracking yaw, pitch, roll, altitude, atmospheric pressure and much more. Wearable technology is related to both ubiquitous computing, and the history and development of wearable computers. Wearables make technology pervasive by interweaving it into daily life.

These smartwatches are a companion and in some cases independent devices from the popular smartphone. Their functionality extends far beyond telling the time. We use our smartwatches as media consuming devices, remote controls, wireless triggers, gaming systems and communication devices. Because of this, the use cases of a wristwatch has grown immensely and consumers are eagerly expecting new methods of interaction and new relevant use cases to be discovered.

1.1 PROBLEM AREA

As mentioned in the introduction, the smartwatch has become almost as important as the smartphone, by providing similar functionalities in a smaller, wearable form factor. While entering the *Internet of Things* era, people are using more and more wearable devices. These devices typically rely on touch as their primary input modality. Because the smartwatch must convey to strict display size constraints, the device is naturally prone to occlusion or the *fat finger problem* [6]. Due to its restrictive size, the information spaces often exceed the limits of the display. Thus, the display acts as a viewport in which the user can bring on-screen information residing off-screen. Users commonly accomplish this by translating the information space into view by touch input. But different interaction methods have been investigated such as external pointing devices [7-12], analyzing built-in sensor outputs (accelerometer, gyroscope, microphone etc.) [13-16], using magnetic field sensors [13, 17], experimenting with real-time image processing [12, 18], using sonar[16] or incorporating external infrared sensors [19-21].

As an alternative to on-screen gestures, researches have investigated peephole displays where the mobile device is aware of its surroundings by "knowing" it's relative position in a predefined workspace[22-26]. With a peephole display the content shown on the screen is updated per the device's movements. Fitzmaurice et al. introduced the concept of the so-called *dynamic peephole*. In this technique, the viewport is physically moved above the virtual workspace that is static with respect to an external frame of reference (see Figure 8). With the help of a spatially-aware display, it is detected which part of the virtual workspace should be shown. The sensors embedded in current devices, such as gyroscopes and accelerometers, would allow such devices to track their relative position, thereby enabling them to become spatially-aware.[24]

1.2 CONTRIBUTION

I believe that current interaction models for handling off-screen information that are used primarily in smartphones (scrolling, pinching, dragging using the device's built-in touch screen) have certain limitations and should not be blindly transitioned to a smartwatch interface. Display size constraints on smartwatch displays are forcing researchers to think outside the box in how users can manipulate data that resides outside the screen space[7-12].

The main goal of this thesis is to find an acceptable interaction model that can improve content consumption on smartwatch devices without compromising on user performance and task completion times. In the following chapters, I will be comparing and investigating current research concerning user interaction on small-touch based devices; constructing, analyzing and comparing different interaction models by combining experimental modeling with empirical data gathered from my experiments. The prototype will undergo an evaluation through real user trails. The evaluation will provide some insights on the envisioned interaction model for creating a non-occlusive input model and more accurate content manipulation with a significant increase in task completion times.

1.3 THESIS OVERVIEW

In this thesis, I investigate the benefits and limitations of interacting with the off-screen space of a modern wrist-worn smartwatch. For my thesis I will be using a motion capture system that would track the position of the smartwatch in real world space. Of course, it's not possible having a full motion tracking camera system attached to the device, due to the technology limitations. The idea behind this approach is that, in the future, tracking technology would have evolved to a point where we will no longer need a room with multiple cameras to enable devices to be spatially aware. The results from the experiments that have been conducted would be used, in theory, when the tracking system technology would be incorporated in the current smartwatch form factor without taking up extra space. For the interaction model found, the future tracking technology should have comparable precision and performance with the current motion tracking system used in the experiments.

I have split this paper in seven main chapters. The second chapter is a review of other interaction technologies and methods already carried out by other researchers. I found out that the research on off-screen implementations are rather scarce for smartwatches and the limited solutions found are not entirely experimented on smartwatch devices. Prior research analysis will give me a starting point in comparing my solution to other researchers. A comparison and summary of different ideas will be presented in this chapter, as well as some comments that might prove valuable for my experimental setup.

The next chapter will contain a detailed and thorough walkthrough of the concept, design and implementation of my prototype and experiments. I will also talk in detail about the challenges I have faced, drawbacks, inconsistencies, and discuss my proposed solutions to solving different problems.

In Chapter 5 I will focus on the user experiments and the interpretation of the results. The experimental setup will be analyzed using industry known methods and techniques. This section will also contain the results of the user experiments, by processing the data, surveys and empirical observations. The evaluation will end in a discussion on the ergonomics of the setup as well as additional comments and feedback from users.

In the last chapter I will provide the conclusion of my thesis with a discussion on whether the problem statement has been addressed or not. Here I will tackle the possibility of improving my model, future work and future research ideas, based on what I have learned from investigating this field of research.

2 RELATED WORK

In this chapter I will describe some of the related research that has been done in respects to interacting with off-screen space on small screen devices such as smartwatches and smartphones. First, I will note some of the problems related to interacting with these devices using different technologies, particularly problems related to mid-air and touch interaction.

Little research has been done in analyzing the way people interact with off-screen space on smartwatches (and smartphones as well), but some researches have been experimenting with the idea of expanding the information space beyond the screen using intricate and innovative technologies. The insights gained from studying the literature helped me design my interaction methods for conducting the experiments. The focus was to find out how current research attempts to solve some of the problems that come with extending the "touch area" of the display.

Because the smartwatch related literature is so scarce I will try to objectively adapt some of the methods used on PDAs and smartphones to my device's screen size and try to create an interaction model based on these research findings. These mobile devices share multiple familiar characteristics such as UI navigation flows and touch input interaction. The literature that investigates PDA and smartphone interaction is relevant for this thesis because of those similarities. The smartwatch is unique to the others by being small enough to be worn on the wrist. The main input method is still touch based and can only be accomplished by using the other hand.

2.1 PROBLEMS WHEN INTERACTING WITH SMALL SIZED DISPLAYS

Hasan et. al [26] compared the direct off-screen space pointing, peephole, and flick & pinch interaction for map navigation tasks and they have identified the following challenges to manipulating content on a smartphone's device:

2.1.1 FAT-FINGER PROBLEM

Devices that incorporate a touchscreen capture a contact point to which it is referred as a 'contact point' This is actually a relatively large 2D region which is converted into a (x, y) pixel coordinate for adding compatibility with traditional pointing models that assume a single point of interaction. However, there is no guarantee that this single point is the true contact point intended by the user. Even though many technology advancements have been made, this problem persists in part due to a lack of sophisticated sensing techniques, but also because the intended point of interaction is inherently ambiguous. Techniques such as "focus+context" lenses have been designed to help mitigate this problem within information visualization applications, but no general solution exists for all types of applications.

Sensing limitations may give rise to a variety of issues when manipulating objects, such as unintentional movement of the object. For example, when a finger lifts up from the screen its contact area changes shape, which may result in a change to the calculated pixel coordinates. If a user was attempting to precisely place a digital object, this might cause the object to shift from the desired target. There has been work to resolve *the fat finger problem* by obtaining a more accurate touch contact point, providing feedback to the user about the success/failure of the touches, and incorporating selective zooming. Although these techniques can provide a more accurate touch contact location, they do not assist substantially in object alignment tasks.[27]

In Siek et. al's research paper[6] they have investigated the *Fat Finger issue* and found out that the older male participants voiced concerns about how their "fat fingers" may cause problems when completing the button press task. They worried that the size of their fingers would cause them to push multiple buttons at the same time or occlude most of the interface. However, they did not find any significant evidence to support this concern during the test.

2.1.2 View size and pointing difficulty

The size of documents computers are able to handle has increased considerably over the last two decades. It is quite common today, even with an inexpensive desktop computer, to have access to huge documents like a complete world atlas. Yet, the size of our screens has not increased, far from it. Screens have remained about the same size, but modern GUIs encourage the opening of multiple windows, at the cost of actually reduced views. Also, research on mobile computing has produced a whole family of devices, like PDAs or cell phones, that offer drastically reduced views. So view size has actually tended to shrink, leading to an increasing mismatch between the size of the information worlds users interact with and the size of the views they enjoy into these worlds.[28]

2.1.3 Input method switching

Many small devices use different methods when interacting, from touch-input, pressing a side button, to using image recognition software[18], or using sonar[16]. This constant switching carries overhead on the performance of completing tasks or time spent accessing information beyond the screen. Some researchers have investigated and devised multiple ways for solving this issue, by committing to using as few interaction methods as possible[7].

2.1.4 Task management

By having such small screen real-estate, the users face a hard choice in how they manage the content on the screen. For example, they can only show only one small portion of a map on the screen, but on a bigger display such as a TV he/she can retrieve much more information at one glance. On a small screen the users must use multiple taps, swipes and view switching in order to complete a certain task. On a bigger screen however, the user is unburdened by the chore of accessing different master/detail views[1].

I did not consider this problem relevant for this thesis, as it is not related to the interaction methods per se, but rather specific to GUI design for small size displays in general.

2.1.5 Occluding UI elements

Adding multiple UI elements can lead to occluding valuable screen real estate. These UI elements must be thought out in such a way that they do not interfere with the user's focus on the task.

[1] A master area can be a form, list or tree of items, and a detail area can be a form, list or tree of items typically placed either below or next to the master area. Selecting an item from the master list causes the details of that item to be populated in the detail area.

Since this is more of GUI design problem, I did not consider it to be very relevant for my thesis, but I will include a small side experiment by adapting the Halo[29] interface for the smartwatch.

2.1.6 Touch input

The current generation smartwatches usually come with a touchscreen with very few physical input buttons. Some smartwatches don't have buttons at all. This ubiquitous method of interaction causes some issues when interacting with the small screen, therefore it is obvious that I start looking into how other researchers have dealt with them. As I previously mentioned, literature on smartwatch interaction methods is limited, so I will include data from devices with similar characteristics such as smartphones and PDAs.

Touch input is the most common method of interacting with small sized displays. Therefore, every task is bounded by touch interactions (scrolling, zooming, panning, taping). These touch interactions can grow in number at a very high rate depending on the task the user has to complete. Touch also has difficulties targeting very small objects and widgets, which is due to the fact that a human finger is quite big, it can occlude the target and it is difficult to assess where you are precisely touching. Finally, touch may introduce general wrist and finger fatigue, given that many operations require multiple hand and even arm movement.

These problems are very important for this thesis, since they describe some of the inherent problems with touch input that I aim to mitigate by combing it with mid-air and off-screen interaction.

2.2 Mid-air and off-screen interaction

In this section I describe some of the mid-air design suggestions that I found in the literature. I have specifically used this advice when designing my own interaction method, but it also initially served me as a way of identifying some of the inherent problems with mid-air interaction.

Vogel and Balakrishnan [30] has described five design characteristics of mid-air pointing devices.

Accuracy

The biggest concern in my model is the user's pointing accuracy. Hand jitter causes a loss of accuracy, but by implementing different visual aids and prediction algorithms, there is a chance that the movement distortions would be minimal. Also due to the fairly small screen, the technology that I use to capture the user's motion must have millimeter precision. When dealing with such a sensitive system, a filtering algorithm is usually needed for creating a much more fluid and predictable interaction. I will go more into depth about input filtering in chapter 3.7.

Acquisition speed

Another important characteristic is how easy and fast it is to interact with the devices using our current model. The device is not used continuously, so there are idle time windows in which the device should be put into an inactive state. Therefore, it is important that once the user starts interacting again, he/she can easily acquire the device and start using it immediately.

COMFORT

Reducing the amount of fatigue is a priority when implementing the new interaction model. Because the user has to interact with the device off-screen, the user cannot rest his finger and has to always keep track of his mid-air finger movements.

POINTING AND SELECTING EASE OF USE

It should be fairly easy to interact with the device using off-screen tracking. The device should work on all distances. Preferable it should preserve touch interaction, and make the transition between touch and mid-air pointing as smooth as possible.

2.3 INTERFACE SCHEMES FOR CONTEXTUAL AND FOCUSED VIEWS

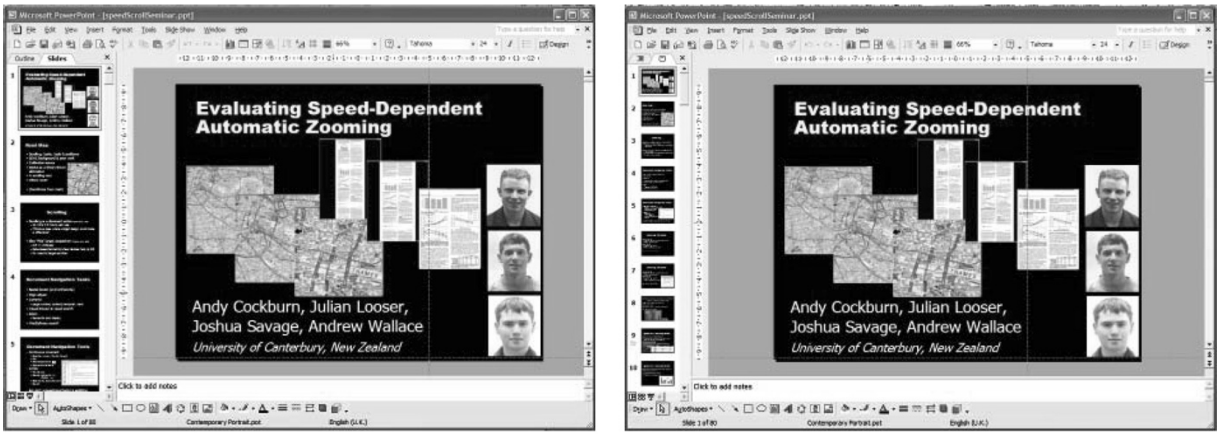
Because the smartwatch display acts as a viewport onto a larger information space, parts of the information space will reside off-screen. Users can access off-screen content through pan-and-zoom interfaces, where they must integrate overview and detail over time [31] or by using world-in-miniature representations of the entire information space [32].

Of course, navigating through these interfaces is harder than it seems. Therefore, proxy-based interfaces have been implemented to assist presentation. Halo[29], Wedge[33], and City Lights[34] are examples of such interfaces, which provide on-screen visual cues of an object's distance and orientation.

Because of the devices screen constraints, it comes only natural to devise an interface where the user can access off-screen space with as little interference as possible and providing enough information so that he/she can carry out the task at hand. A paper by *Cockburn, Karlson and Bederson[31]* review and categorize such schemes according to the interface mechanisms used to separate and blend views. They analyze four interface schemes that allow users to obtain both focused and contextual views of their information spaces.

2.3.1 OVERVIEW+DETAIL: SPATIAL SEPARATION

As stated in the paper by Cockburn et al., an *overview+detail* interface design is characterized by the simultaneous display of both an overview and detailed view of an information space, each in a distinct presentation space. An example of an interface with an *overview+detail* implementation can be seen in Figure 1.



(a) five thumbnails in the overview (b) ten thumbnails in the overview

Figure 1 PowerPoint's overview+detail interface. The scrollable thumbnail overview is on the left-hand side of each window, and the thumbnails can be scaled by configuring the width of the overview region.[1]

This immediately raises the problem of screen real estate for the small smartwatch screen because if I intend to use this interface scheme as described, I would need to reserve a portion of the screen for the overview panel.

Due to the physical separation of the two views, users interact with the views separately, although actions in one are often immediately reflected in the other. Also, this would not help in any way in the user's navigation for a smartwatch. The screen is too small and the user needs to have a more direct feedback during the motion of panning through virtual space.

Scrollbars and thumbnail overviews are not necessary for this task. They are a perfect fit for desktop applications such as PowerPoint or Adobe Reader, but in my case, they offer little to no direct aid in completing the task.

2.3.2 Zooming: Temporal Separation

Another category of interface supporting focused and contextual views is based on *zooming*. This involves a temporal separation between views. The user could zoom in and out of context to better navigate through the interface. I have considered magnification for helping the user navigate and scroll faster to the target's location, but ultimately dismissed it, due to the unnecessary added complexity for implementing clutching strategies when transitioning from panning to zooming. Implementing a zoom mechanism rises several issues. Because I am tracking only one marker point that represents the user's finger, the clutching between panning and zooming becomes difficult. This adds unnecessary overhead to the prototype design and does not help in establishing a one-to-one comparison between *touch* and *mid-air pointing*. Users might use the zoom interface more when it comes to the touch input method because of the relative ease of use, while as in mid-air pointing this type of interface might become cumbersome and might not be used as much. Therefore, I have decided to not use this type of interface in my design. It also could introduce more user errors such as zooming in by mistake. Zooming too far or too close have a moderate chance of happening and it would be wise to avoid these events, if possible.

Another potential limitation of combined zooming and panning is that it may impede users in forming and exploiting a spatial understanding of the information space. The data moves to different locations in the screen, dependent on space and scale settings.

2.3.3 *FOCUS+CONTEXT: SEAMLESS FOCUS IN CONTEXT*

Focus+context is another approach in which all parts are concurrently visible: The focus is displayed seamlessly within its surrounding context. Research into fisheye view interfaces, which diminish or suppress information that lies away from the focal area, is particularly well represented in this category.

Two potential problems with fisheye views are, firstly, misinterpretation of the underlying data, and secondly, challenges in target acquisition, both of which are caused by distortion of the information space. This would impede the ease of use on the target's acquisition process because the participant might not notice the target in the distorted area. Limited screen space is also a concern here.

2.3.4 *CUE-BASED TECHNIQUES*

Almost all of the *overview+detail*, *zooming*, and *focus+context* approaches described earlier modify the size of objects to support both focused and contextual views. These scale modifications can be applied purely to the graphical portrayal of objects, or semantically so that only objects with certain properties are scaled. *Cue-based techniques*, on the other hand, modify how objects are rendered and can introduce *proxies* for objects that might not be expected to appear in the display at all. I consider this method to be the most promising out of all described earlier. The size constraints of the smartwatch's display are not a factor in acquiring the target's location on screen.

For my experiments, I drew inspiration from a cue-based system named Halo[29] by which offers contextual cues about information lying outside the main display region. In Halo (Figure 2), a variation of City Lights [34], Baudisch and Rosenholtz explored the use of arcs as decorators, as though each off-screen object were a street lamp just tall enough to cast its circle of light into the screen viewport. Nearby objects cast short, highly curved arcs, while far objects cast long, subtly curved arcs. In this way, object direction and distance are encoded as arc placement and curvature.

Figure 2 A mapping application implementing Halo for displaying information about off-screen targets [29]

I investigated interaction techniques for off-screen content that are relevant to designing the prototype that I will be using when retrieving data from my experiments. Irani et. al has some interesting ideas when designing these types of interface models and validates that proxies are useful for accessing off-screen elements [35]. However, these techniques require that the user has knowledge beforehand when navigation through off-screen content.

I did not implement Halo, but adapted the cue technique to a simple target pointer that translates and rotates towards the target's location while the user is panning. An example of my implementation can be seen in Figure 16. I will discuss thoroughly about my design choices in chapter 3.11.

2.4 Interaction with small devices using external or internal sensors

This section investigates other novelty interaction models that have been developed with the use of sensors that are built-in, or add-ons for extending the device's spatially-aware capabilities. These models should offer some insights on how I should design my prototype for handling off-screen information space.

2.4.1 Abracadabra

The technology entitled "Abracadabra" proposed by Harrison and Hudson in 2009 [17], uses a magnetically driven input approach that makes use of the larger space around a very small device (for example a smartwatch). The researchers used a multi-axis magnetometer, which is often used to determine the orientation relative to the Earth's magnetic field. They can take advantage of this capability by strapping a local magnet to the user's finger. This overrides the Earth's magnetic field and makes it feasible for the user to interact with the device. The finger acts like a wireless input device and taking advantage of the relatively larger world space around the display.

Figure 3. Prototype Abracadabra-augmented wristwatch.[17]

2.4.2 GestureWatch + Hoverflow

Another external sensor based solution for taking advantage of off-screen space is the GestureWatch [36]. The GestureWatch is a gesture based wireless contact-free wristwatch interface. It uses a set of five infrared proximity sensors to detect a variety of hand gestures. It is important to note here that this technology utilizes only the space above the device, thus not taking advantage of the off-screen space.

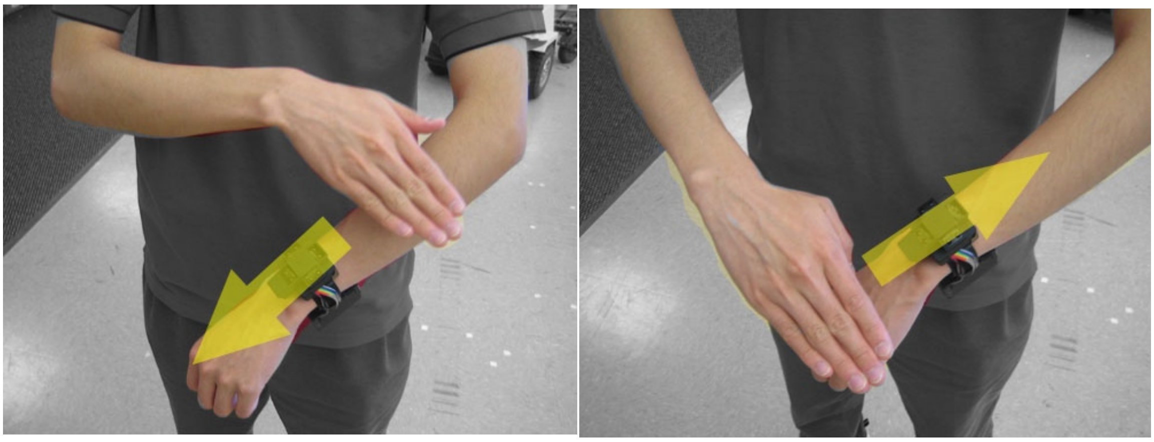


Figure 4 The Gesture Watch prototype that uses external sensors for gesture recognition. [36]

The Hoverflow system [20] is similar to the GestureWatch implementation by using six infra-red sensors mounted on the device. Hoverflow allows the user to select colors from a color palette through hand gestures. Possible gestures are moving the hand across the device, presenting several hand postures, or by moving a hand rapidly towards or away from the device.

2.4.3 Surround-see

Surround-see [18] enables peripheral vision around the device to augment daily mobile tasks. Surround-see provides mobile devices with a field-of-view collinear to the device screen. This capability facilitates novel mobile tasks such as, pointing at objects in the environment to interact with content, operating the mobile device at a physical distance and allowing the device to detect user activity, even when the user is not holding it. The future work for this type of technology will hopefully enable the device to have full 3D depth sensing capabilities. In theory, this technology will be used in collaboration with the ideas presented in this thesis for enabling full off-screen interaction.

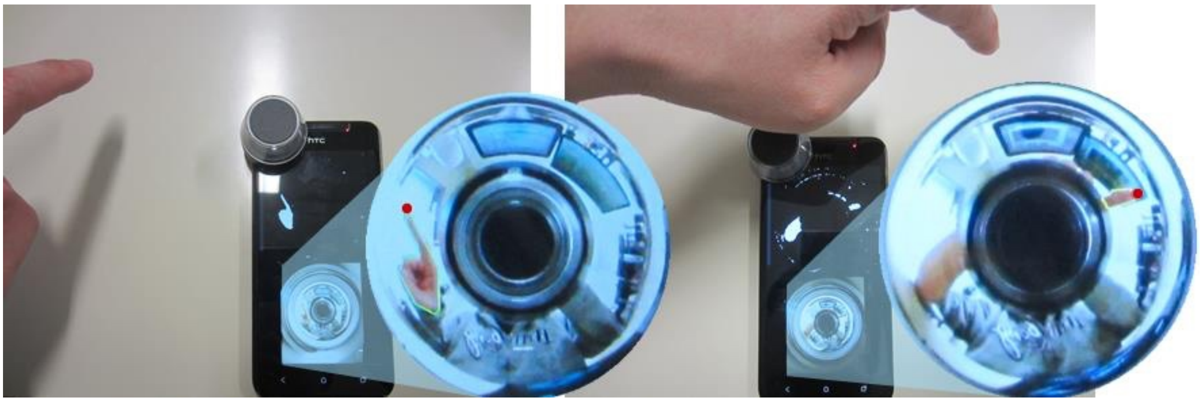


Figure 5 Surround-See enables peripheral 'sight' on smartphones by means of an omni-directional mirror attached to the mobile device's front facing camera.[18]

2.4.4 *PORTICO*

Portico [10] explores the off-screen interaction space that I am interested in. The physical design of the system with two cameras enables it to view and respond to users' interactions with objects on the surface surrounding the device. Thus, the system's input space (what the system sees) is significantly larger than its output space (limited to the screen boundaries). The proof of concepts presented in this paper show that this type of interface can be used in the "real world" and highlight the potential benefits of including around device sensing technologies in small or large portable devices.

Figure 6 The Portico system in use. Two cameras track objects on the screen and surrounding surface. In this application, a toy zebra is tracked.[10]

2.4.5 *SIDESIGHT*

SideSight [21] is part of the same category as GestureWatch and Hoverflow by integrating infra-red proximity sensors. The difference here resides in the placement of the sensors. They have been embedded along each side of the device and they are capable of detecting the presence and position of fingers in the adjacent regions. Unlike GestureWatch and Hoverflow, when rested on a flat surface, the user can carry out single and multi-touch gestures using the device's off-screen space. This is relevant for my thesis' research because it gives some insights on using a larger input space instead of just the on-display touch input.

2.4.6 FINGERIO

FingerIO [16] uses a modulation technique commonly used in wireless communication called Orthogonal Frequency Division Multiplexing (OFDM). It transforms the device into an active sonar system that transmits inaudible sound signals and tracks the echoes of the finger at its microphones. This alleviates the problem of having some external based sensory tracking system and use the internal microphones that a smartdevice usually already has. The smart watch form-factor fingerIO prototype presented in this paper shows that it can extend the interaction space to a 0.5×0.25 m^2 region on either side of the device and work even when it is fully occluded from the finger. In comparison, the solution proposed in this thesis, would theoretically allow a virtually unlimited interaction space across all spatial axes.

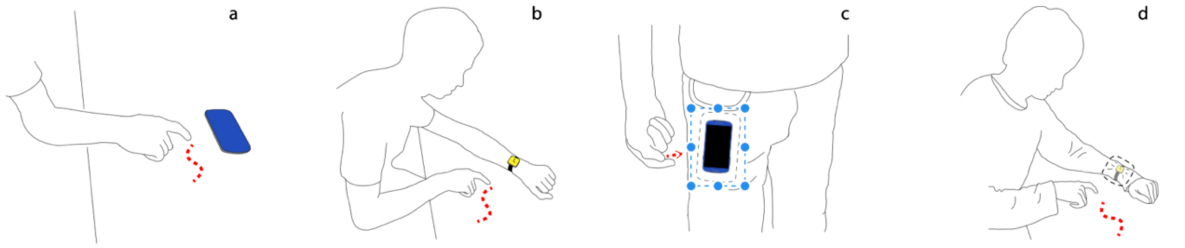


Figure 7 Applications of fingerIO. a) Transform any surface into a writing interface; b) provide a new interface for smartwatch form factor devices; c) enable gesture interaction with a phone in a pocket; d) work even when the watch is occluded. [16]

2.5 OFF-SCREEN INTERACTION

Due to scarce research that has been made for mid-air pointing for small screens such as those used in smartwatches, I have decided to investigate and include other mid-air interaction models (wall-sized displays for example) to see if some methods might be relevant for this thesis. This section investigates how other researchers have tackled modelling off-screen interaction on smartphone, wall-sized, tablet or smartwatch displays.

A very relevant study for this thesis has been the work conducted by B. Ens et al [37] that investigates the ability to directly point at objects that are outside the viewport. They have stated that direct input techniques involve a one to one correspondence between the location of the input device (such as a finger) and the location of the virtual object. The term direct off-screen pointing describes the ability to directly select objects that lie outside the viewport. Direct off-screen pointing can allow familiar item locations to be specified without visual feedback, but on-screen targeting assistance, via visual cues for off-screen targets, or off-screen target cues, are required for items whose whereabouts are previously unknown.

The researchers have concluded with some guidelines for designing interfaces for directed off-screen pointing. They suggest using different cue-based implementations for close or distant targets. They state that cues for off-screen target should be carefully selected as their scent capacity can vary with distance; a cue well-suited for shorter distances could impair performance at larger distances. Because this thesis is not investigating cue-based systems, I will consider using only static cue-based visual aid for both close and far targets.

Hassan et al. assumes that 3D finger tracking will become possible [38]. This thesis is based on a similar assumption, and investigates how the user's would interact with similar interfaces like AD-Binning. In their first experiment they noticed that "tapping" and "dwelling" on targets had the least errors and a trend toward faster selection times, but they also noted that these may not be practical in all applications. For example, dwelling may conflict with object browsing, and tapping should only be restricted to a specific on-screen target. In the second part of the same experiment, they concluded that selecting targets in the closest distances was less accurate and less efficient. Due to constrained movements with crossing arms, areas left and top-left of the device are generally more cumbersome. The second experiment showed that interacting with a smaller item group size provided more flexibility regarding bin choice, but participants clearly avoided using the left and top-left sectors.

Other researchers have been comparing different interaction methods in terms of accuracy, speed, expressivity and effort [39]. These experiments have been conducted without visual aids. There was no notion of hitting or missing the target. Instead, every selection provides data on selection accuracy (the distance between the ideal target location and the user's estimation). Although, unlike traditional target acquisition, this important experimental design decision allowed for characterizing the underlying human factors of spatial target acquisition with the different interface styles, whereas traditional discrete targets would not, because there would be no basis for selecting any particular number or layout of targets in each spatial arrangement. The 2D plane interface generally outperformed the others. Participants were able to make rapid and relatively accurate selections, even in the absence of any guiding feedback. Raycasting allowed similar acquisition speeds, but with much less accuracy; however, it was rated as the least physically demanding and would be a good pointing solution for rapid coarse selections where there are few candidate targets. All of the interfaces suffered from drift, with selections progressively moving further from targets when no feedback was provided. Also to note here, participants had particular difficulty acquiring targets on the 3D depth plane. This outcome impacted the design decision of working only in a 2D plane for my prototype, on which I will elaborate in the next section.

Kerber et al. [24] didn't investigate mid-air pointing per se, but instead took a different approach by moving the device altogether in report to world space. They developed a dynamic peephole interaction technique where the user controls the position of the peephole with his arm movements. By moving the arm in 3D space the user can interact with the device without actually touching the display. They reported that the dynamic peephole interaction performed significantly worse in terms of task completion time. The 12% higher task completion time compared to direct touch interaction would argue against using it. Nonetheless, the advantage of exploring large-scale content with one hand might compensate for the slower speed.

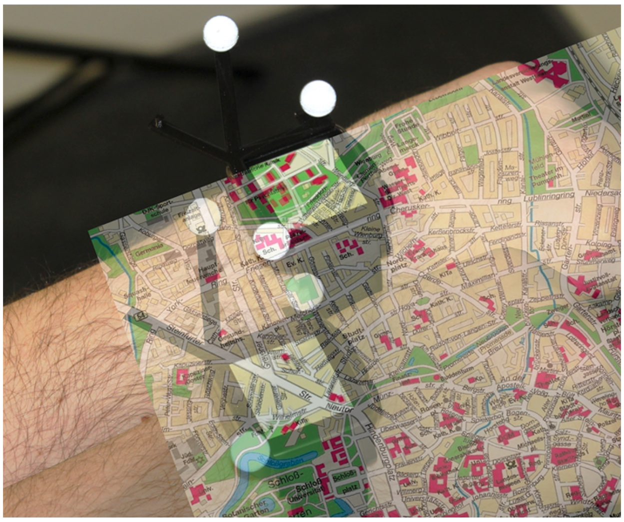


Figure 8 A dynamic peephole map application - The user moves his arm to browse the map (created by blending two photos and the original map)[24]

A paper describing an user study evaluating mobile free-space navigation techniques and the impacts of sensor orientation on user experience, found that free-space interactions were found to be more fatiguing and users would be less likely to use them in public settings.[40] Constraining the working interaction volume greatly impacts user experience, with tradeoffs between sensor orientations. Nonetheless, the main finding of the study was that a free-space technique performed equivalent to touch screen interactions. This supports the idea that interacting beyond the device's display would not impact negatively the current status quo interaction method, namely touch. Also in their work the results showed that for multiscale navigation, the majority of users preferred free-space interactions to touch screen, with equivalent movement times.

In a study conducted by K. Matsumura[41], he has investigated user-defined gestures on the surrounding area of the screen through a user study using paper prototypes. They realized that people often point or gesture not only on the touch-sensitive screen of a device but also around the screen. They discovered different off-screen gestures and interactions that can be used in designing real world prototypes. In his findings he observed that that it is important to include an action and/or a voice command to express an explicit period for the gesture. This implies the use of the microphone and voice recognition technology may be required to realize the off-screen interactions. This would remove the necessity of developing advanced clutching techniques. He proposed some different user-defined gestures are shown in Figure 9.

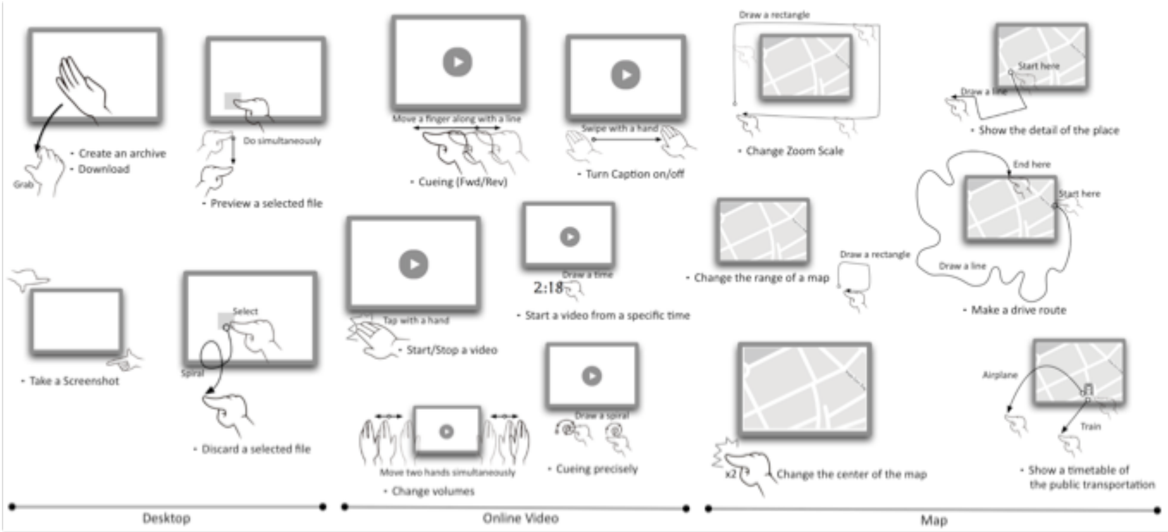


Figure 9 Examples of user-defined gestures for each content. [41]

Knibbe et al. extends the interaction space to the side of the watch using infrared sensors mounted on the side of the device.[19] Their prototype supports multiple gestures such as single finger taps, multi-finger pinch-to-zoom, hand waving etc. They stated that the user feedback was positive; the larger interaction area was found to be beneficial supporting the points raised in their related work, and the use of bimanual gestures were described as 'natural' and 'intuitive'. The prototype device demonstrates the possibility of bimanual gesture input to minimize screen occlusion and maximize the interaction surface.

Finally, a paper, that had a relatively important impact to my thesis, is a paper by Markussen et al. which investigates *Off-Limits*, an interaction concept extending the input space of a large display into the space beyond the screen through the use of mid-air pointing.[42] They conducted three empirical studies and compared touch interaction with mid-air pointing. Their data suggests that mid *Off-Limits* concept provides significant performance benefits over on-screen and touch pointing conditions, but this is relevant only for large displays. The question here is how well would a similar technology work on very small displays.

2.6 DISCUSSION AND SUMMARY

When including the different choices in my final design proposal, I took into account what I wanted to investigate. I wanted to exploit the idea of using the off-screen space as a means of controlling different applications such as scrolling through lists, using a map application etc. The performance, accuracy and ease of use are just some of the criteria that my proposed design will be graded on.

After analyzing the aforementioned studies and articles which tackle some kind of mid-air pointing, handle off-screen interaction or design interfaces that are suitable for off-screen use, I began analyzing the usefulness and quality of each technology and the possibility of completing a porotype for gathering data. Due to time and human resources limitations, I had established the following design guidelines for my prototype.

I have discarded all ideas that included external sensors or pointing accessories. I considered mid-air pointing to be the best candidate when it came to enabling the users to interact with the smartwatch's small screen. It offered little to no occlusion when manipulating the device's interfaces and had stable results when it came to accuracy and performance.[37] Other researches noticed that limited ease of use and awkwardness was something that the user's had to deal with in order to adopt this type of interaction model.[40] This thesis is also investigating the degree of comfort and ease of use when using mid-air pointing on smartwatches. For example, assuming the users are wearing the watch on their left wrist, how difficult would it be to extend the right arm to point and drag from the left side of the smartwatch?

In the AD-Binning experiments, dwelling on targets had the least errors and a trend toward faster selection times, therefore it gave me confidence to adapt this particular method in my empirical study.

The interface for my study is also important. From the different versions presented earlier, I have adopted a cue-based system, because it uses just enough screen space so that the user can navigate and locate the target when scrolling. I consider it to be the best approach for completing the tasks at hand.

Privacy and social acceptance will also impact the design of the my study. How accepted is my solution in terms of social interaction? Do users feel awkward or not when there are other people around them during an interaction. Extending the arm fully is more or less embarrassing when trying to use only a small part of the off-screen space?

3 DEVELOPMENT OF THE MID-AIR TRACKER

In this chapter I will be discussing the design of my prototype and the development steps taken to create the mid-air tracking system. I decided to evaluate the panning motion mentioned in the previous chapter, by comparing the mid-air approach with the currently ubiquitous interaction method present in almost all smart devices, touch. Panning is one of the most important interactions when accessing off-screen information. I extended the comparison even further by enabling inertia for the touch model to have a more realistic user scenario.

3.1 TRACKING OFF-SCREEN MOTION

Different technologies exist for tracking the user’s motion, but few offered the precision and latency needed for carrying out my experiment. In order to capture the user’s movement I had to look for technologies that enable the prototype to record in real time the movement of the participant’s smartwatch, worn on the wrist, and his/her fingertip. After comparing the pros, cons and availability of these technologies I have decided to opt for a professional external optical motion tracking solution that could record the positions in world space at a high bitrate.

An optical motion capture system utilizes proprietary video cameras to track the motion of reflective markers (or pulsed LED's) attached to particular locations of the subject. I use a reflective optical motion capture system that uses Infra-red (IR) LED's mounted around the camera lens, along with IR pass filters placed over the camera lens. Optical motion capture systems based on Pulsed-LED's measure the Infra-red light emitted by the LED's rather than light reflected from markers. The centers of the marker images are matched from the various camera views using triangulation to compute their frame-to-frame positions in 3D space. If two calibrated cameras see a marker, a three-dimensional fix can be obtained. Typically a system will consist of around 2 to 48 cameras. Extra cameras are required for full coverage around the capture subject and multiple subjects. In my setup, I have used 12 cameras to track the smartwatch and finger position.

3.2 CURRENT SETUP

For my experiment I used 12 OptiTrack’s Flex 3 cameras that are capable of capturing fast moving objects with its global shutter imager and 100 FPS capture speed. By maximizing its 640 × 480 VGA resolution through advanced image processing algorithms, the Flex 3 can also track markers down to sub-millimeter movements with repeatable accuracy which is relevant for tracking the most minute details in the participant’s motion. OptiTrack’s Motive application helps avoid several problems that often plague passive optical systems, including swapping of markers, noisy or missing data and false reflections.[43]

Flex 3 cameras emit 850nm IR light, which is nearly invisible, for inconspicuous and gentle lighting, thereby avoiding vision fatigue and unwanted attention to your capture rig caused by cameras that emit bright, visible spectrum light. I used the most precise marker detection option available for the Flex 3 which sends grayscale marker information to the PC for calculation of object location, size, and roundness using precise processing methods. It provided greater accuracy than other modes, but consumed more CPU resources and USB bandwidth.

The smartwatch used in this thesis is a Simvalley AW-414.Go model that has a 1.3 GHz processor and 512 MB of RAM. The watch has a 1.5" touchscreen with a resolution of 240x240 pixels. It has two navigational buttons on the side. It has 4GB of internal memory with the possibility of expanding it with a microSD card. The smartwatch's dimensions are 14.1 x 45.3 x 44.3 mm (height x width x depth), and it weights 91g. The operating system installed on the device is Android 4.2 also known as Jelly Bean. The communication features of the Simvalley Mobile AW-414.Go can also be compared with an entry-level smartphone.[44] The WLAN module supports the IEEE standards 802.11 b/g/n in 2.4 GHz networks. Signal quality is below average, the signal was very weak with a distance of 10 meters from the router (ASUS RT-AC66U, a dual band 802.112AC gigabit router). There was a small noticeable delay during communication with the streaming PC as well, but I will discuss more in detail in a later chapter, when analyzing the limitations of my project.

3.3 Smartwatch & finger representation in 3D space

In order to track the smartwatch's position and rotation in 3D space, I used Motive's rigid body solving feature and 3D reconstruction to represent the smartwatch in virtual space. By having at least 3 reflective markers placed on a rigid plate, that is placed on the back of the watch, I am able to track it's real world position and rotation in real time. By having a minimum of 3 markers for tracking the watch I am able to record not just the position but also the rotation. The smartwatch is free to change position as forward/backward (surge), up/down (heave), left/right (sway) translation in three perpendicular axes, combined with changes in orientation through rotation about three perpendicular axes, often termed pitch, yaw, and roll (see Figure 10).

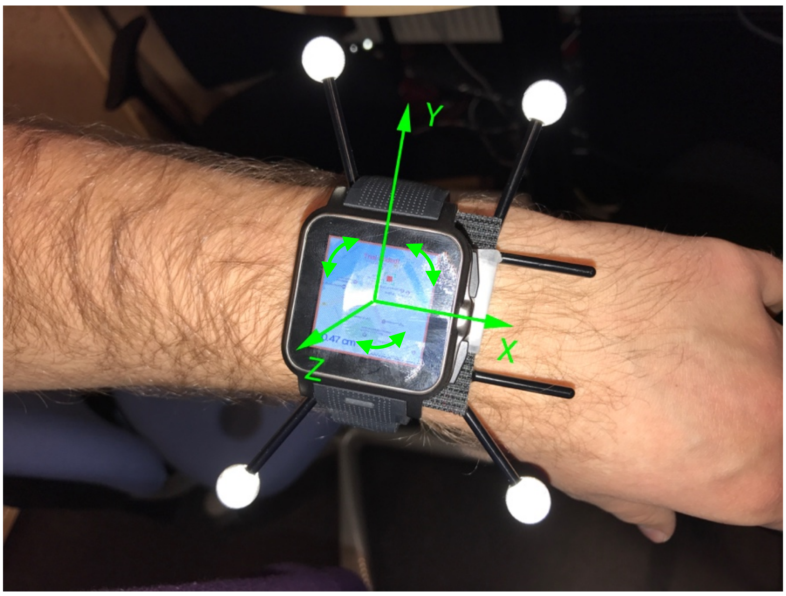


Figure 10 Freedom of movement of the smartwatch in three-dimensional space having six degrees of freedom.

The finger is represented as a single reflective marker that is placed on top of the pointing fingernail, more exactly on the distal phalange. This allows for an accurate measurement of the user's pointing finger position. However, I do not record the pointing angle; this would require at least 2 reflective markers placed on the user's finger. I have decided, after much consideration, to remove this idea because of two main complications. The first one was the time constraint for finishing this project and the second, by recording two markers on the finger, it would further complicate the system and would introduce other tracking problems when clutching between dragging and idle states. When a user wanted to target a specific point I would have had to design a different clutching mechanism to initiate the dragging motion. This can be an excellent point for expanding and improving the system for future work, but would be outside of the current thesis scope.

3.4 ARCHITECTURE

Due to the smartwatch's technical limitations, processing and recording the data streamed by OptiTrack's Motive software cannot be accomplished on the device itself. The data is streamed over the local network using unicast to a Mid 2015 Retina 15-inch MacBook Pro running Windows 10. The laptop has a 2,2 GHz Intel Core i7 processor with 16 GB DDR3 memory. This machine serves as a log recorder and streaming server that processes the data from the OptiTrack server and logs and calculates the positions of the other entities (smartwatch and finger) in real-time. In order to have a direct representation of the world space in virtual 3D space, I am using Unity to calculate and manipulate virtual 3D objects.

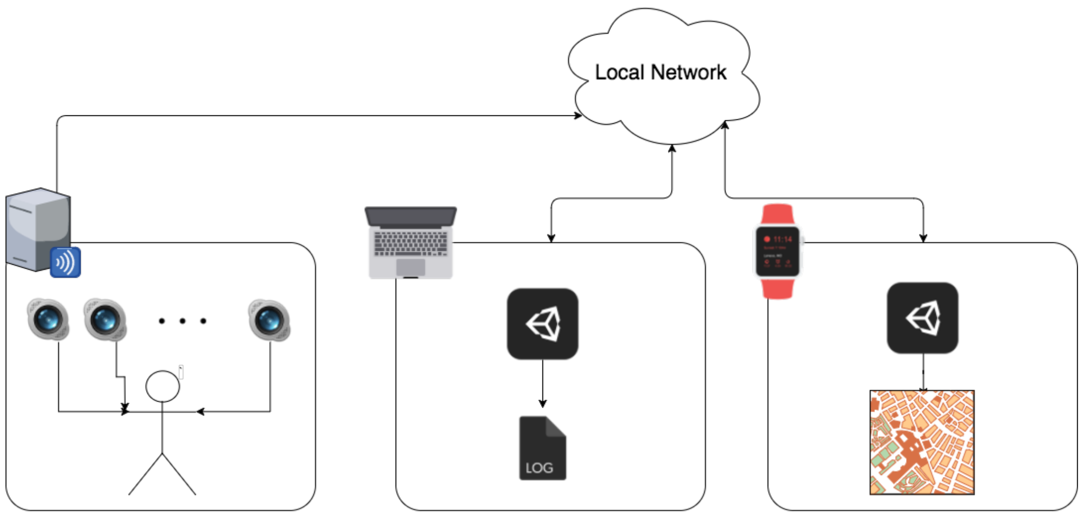


Figure 11 The prototype's architecture where the OptiTrack server is capturing the user's movement, then streaming the data to the laptop to perform logging and all necessary calculations, which then sends necessary information to the smartwatch.

The Unity game engine makes it ideal for processing the raw data from the OptiTrack server. Because the smartwatches hardware is not capable of unwrapping, processing and logging the streamed data. The entire logic is being done in an Unity instance running on the laptop and the smartwatch becomes a remote viewport for the scene.

The connection between the two devices is accomplished by using Unity's Low-Level Transport Layer API (LLAPI). The Transport Layer is a thin layer working on top of the operating system's sockets-based networking. It's capable of sending and receiving messages represented as arrays of bytes, and offers a number of different "quality of service" options to suit different scenarios. It is focused on flexibility and performance, and exposes an API which

I can use. Using LLAPI enables me to build my own networking system at a lower level, which turned out to be very useful in minimizing the latency between the laptop and the smartwatch.

3.5 MID-AIR & TOUCH SCENES

The prototype consists of two scenes: one for handling mid-air interaction and the other for recording direct touch input. These scenes had to be split apart because even though the data processing and logging had to be done on the laptop, the interaction for investigating touch had to be done on the watch. Thus, the watch is now a smaller streaming entity that uses the same LLAPI as in the mid-air experiment.

In the mid-air interaction experiment, the laptop is continuously streaming the positions of the smartwatch, target and finger; it also streams if the finger is inside the clutching area or not (I will be discussing this later on) in order to provide feedback to the user that he is indeed panning within the interface.

During the touch experiment, the watch is streaming the target's position, touch status (idle or panning) and the number of clutches back to the laptop, which is handling data logging. Of course, some packets are dropped during transport, and the laptop is saving only a portion of the data and synchronizes only the last received state from the network. The data streamed back to the laptop is good enough for analysis and should not cause any ambiguous results. The finger's position is tracked using the same method as in the mid-air tracker, by having a reflective marker in the pointing fingernail. I took this decision because it maintains data consistency across the two different experiments.

Unity and Motive coordinate systems

Motive (1.7+) uses a right-handed Y-up coordinate system (Figure 12b) while Unity uses a left-handed coordinate system (Figure 12a). In order to provide consistent positioning and rotation between both applications I had to invert X and W using the following formula:

```
// Flip coordinate handedness from right to left by inverting X and W.
rbState.Position = new Vector3(-rbData.X, rbData.Y, rbData.Z);
rbState.Orientation = new Quaternion(-rbData.QX, rbData.QY, rbData.QZ, -rbData.QW);
```

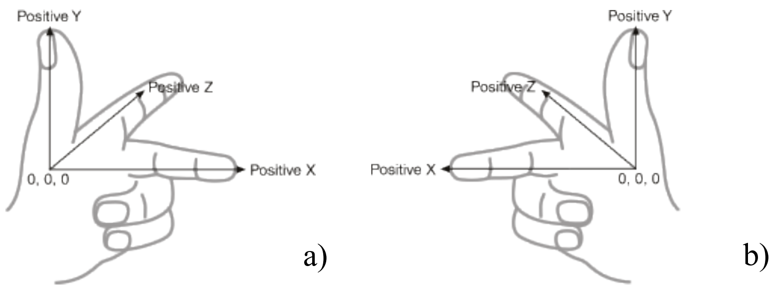


Figure 12 a) Unity's left-handed coordinate system. b) Right-handed Y-up coordinate system for Motive 1.7+.

3.6 FINGER POSITION ALGORITHM

Although the OptiTrack setup provides excellent precision when it comes to calculating a rigid body's position and rotation, it falls short when it has to track single unlabeled markers such as the pointing finger's position. Because of noise and other materials in the experimental room, tracking the unlabeled marker at all times becomes difficult. Another problem arises when a user occludes the other markers attached to the smartwatch when panning. When occluding some markers, the OptiTrack streaming application automatically moves the rigidbody's markers to the "other" list of markers. This list of markers contains any marker that hasn't been assigned to a rigid body. In an ideal environment, there would be only the pointing finger's marker coordinates in this list. Another problem I encountered was that the first appearance of the unlabeled marker would always be present in the "other" markers list, due to a Motive 1.7 bug. Therefore, it made it harder to know at all times which marker is the finger.

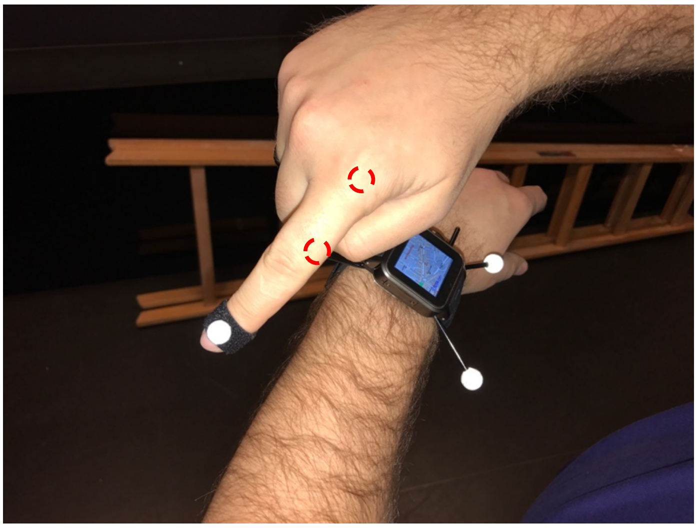


Figure 13 The user is occluding two of the reflective markers that are used to represent the rigidbody in Motive

My solution for these problems consists of ignoring the initial marker and iterating through the list of other markers. An interesting thing to point out is that the finger has a continuous motion during tracking, thus it would make sense to compare the new markers with the positions recorded earlier. The next step would then be to find the closest marker relative to the previous "correct" finger position. I also ignore the last known rigid body's markers' positions because they should not be mistaken for the user's finger. This calculation is executed every frame.

3.7 REMOVING TRACKER NOISE

Noisy signals occur when an original time varying value undergoes undesirable and unpredictable perturbations. Noisy signals are a common problem when tracking human motion, particularly with custom sensing hardware and inexpensive input devices like the Kinect. In addition, even signals from established high-end sensing systems can become noisy when interaction techniques use large scaling effects. Noise affects the quality of a signal in two primary ways. It can reduce accuracy, by adding an offset between the observed values and the true ones. More often, it reduces precision, where repeated observations of values exhibit jitter – many different values are observed for a single true one. Jitter has a large effect on the way people perceive and act.[45]

For this thesis I decided to opt for using the *1€ filter filtering algorithm*[45], a simple algorithm to filter noisy signals for high precision and responsiveness. It uses a first order low-pass filter with an adaptive cutoff frequency: at low speeds, a low cutoff stabilizes the signal by reducing jitter, but as speed increases, the cutoff is increased to reduce lag. The algorithm uses very few resources, and with two easily understood parameters, it is easy to tune. In a comparison with other filters, the *1€ filter* has less lag using a reference amount of jitter reduction.

By filtering the positions and rotations given by the OptiTrack server, the user can interact with an increase in precision and can focus on the task at hand without the need of readjusting his pointing finger or smartwatch. This filter also allow small hand jitters to be ignored while being tracked by the high precision cameras.

3.7.1 Tuning the 1€ filter for my prototype

To minimize jitter and lag when tracking human motion, the two parameters (*fcmin* and *beta*) can be set using a simple two-step procedure. First I set *beta* to 0 and *fcmin* to a reasonable middle-ground value such as 1Hz. I move my finger at a very slow speed while *fcmin* is adjusted to remove jitter and preserve acceptable lag during these movements (decreasing fcmin reduces jitter but increases lag, *fcmin* must be > 0). After I have an acceptable value for *fcmin*, I move the finger in all directions at a higher pace and increase *beta* with the focus of minimizing lag. The authors of this filtering algorithm note that parameters *fcmin* and *beta* have clear conceptual relationships: if high speed lag is a problem, increase *beta*; if slow speed jitter is a problem, decrease *fcmin*. I have found that using a *fcmin value of 0.875* and *beta value of 0.1*, produced the best results for my prototype.

3.8 Abandoning touch without inertia trial

Initially, I had planned 3 type of trials for comparison: *touch*, *touch+inertia* and *mid-air pointing*. It became apparent for me that having a trial investigating *touch* without inertia would only create discomfort, tiredness and annoyance for the user. After conducting a subjective pilot trial to test if including the *touch* without inertia trial would add any new insights. After comparing the results, I had to conclude that adding the *touch* trial in the empirical study would not justify its usefulness in the real world[2]. Users felt frustrated when panning multiple times a target that was 1 meter away. The smartwatch's touch area was way too small to carry out the task in a reasonable amount of time (one of the trials took 27 min to complete) and had to abandon this trial type.

3.9 Enabling inertia for touch

Touch-screens often use inertial scrolling, in which the scrolling motion of an object continues in a decaying fashion after release of the touch, simulating the appearance of an object with inertia. By enabling this feature, the user can access more off-screen information with less touch interactions. The algorithm for enabling inertia for touch can be seen in Appendix B.

3.10 Mid-air touch area

During the prototype's development process, I have iterated through multiple ways of handling input for mid-air pointing, as explained in an earlier chapter. That being said, the objective for this project was to include as little components as possible to the system. Because the use of

[2] The comparison results are available in the archive attached to this thesis.

gyromice and other extra hardware was not recommended, I eventually decided that I should use the finger's relative position to the watch when navigating through the interface.

Using the Unity game engine, I have attached an invisible area to the smartwatch that spans a high enough distance from its center, so the user can target, point and drag from any distance. The shape I chosen for the experiment is a rectangular parallelepiped measuring *10m x 0.2m x 10m* in width, height and length. When the user has his/hers pointing finger in this designated volume, it mimics a pan gesture that translates the target depending on the finger's position relative to the watch.

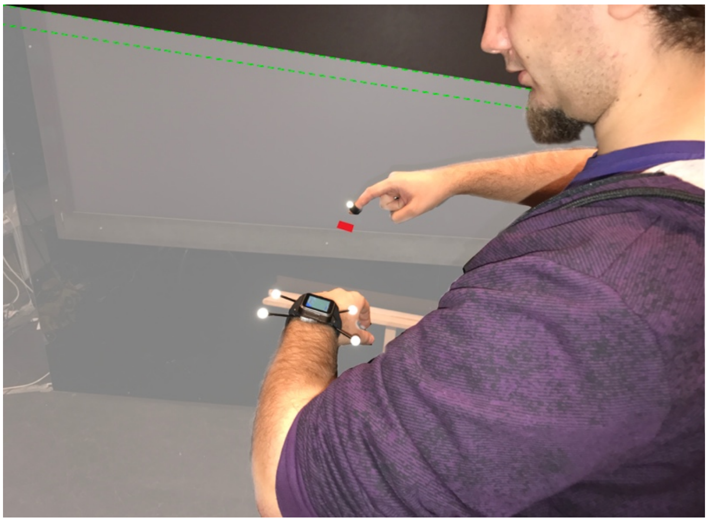


Figure 14 The user moving the target towards the center of the watch while the finger is inside the invisible "drag" area.

The "touch area" is translating and rotating every frame depending on the smartwatch's position and orientation and is listening on the finger's position. This approach made the mid-air interaction more intuitive for the users because they could imagine they can just touch an invisible extended surface.[3] When the finger enters the observed area, the laptop sends commands and positional data to the watch telling it that a pan gesture is occurring.

This approach does have some limitations however. By not having an immediate clutching mechanism that could provide physical feedback (such as a button or switch), the system must present a visual feedback displayed on the watch's screen. When the finger is triggering the drag operation, there is a thin outer border across the entire display that lights up either green or red. The red color represents the state where the user is inside the "touch area", and the gray one represents the idle state (see Figure 15) when the user is out of the "touch area".

[3] Questionnaire answers are available in the archive.

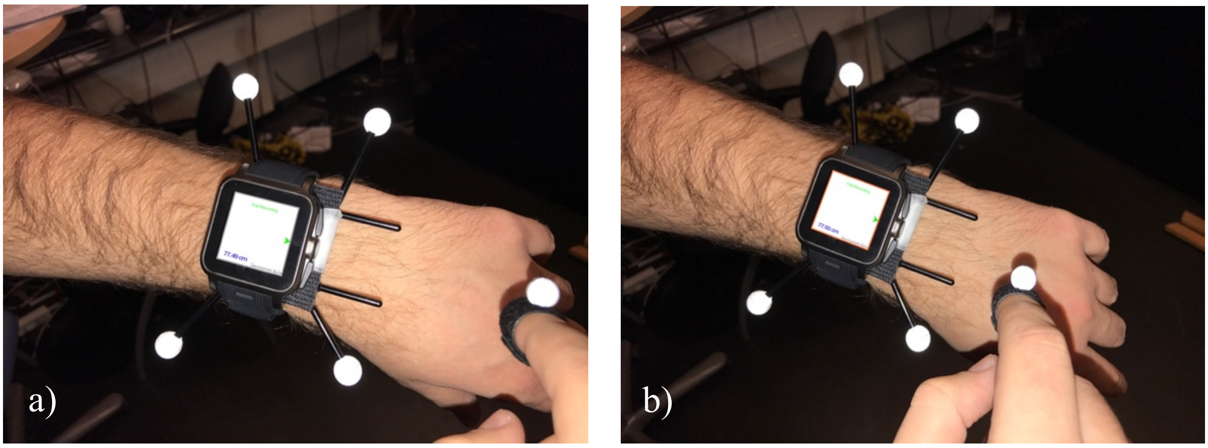


Figure 15 a) Idle state - the user is not panning and the target maintains its current position. b) Panning state - When the user's finger is in the respective "pan area" that is attached to the smartwatch, he/she can freely manipulate the target's location without touching the screen.

Another downside for this method of interaction with the interface is that the users always have to focus their attention on the screen. This of course could be addressed by having audio cues that would emit certain sounds when the pointing finger is in the "touch area". But for the purpose of this experiment, it is safe to assume that the implementation of such cues is not necessary.

3.11 ON-SCREEN VISUAL AIDS

Because all targets are located in off-screen space and the display is the only element that can provide some sort of feedback for the user, I have developed an on-screen visual target tracker that changes position every time the target is being translated. This feedback mechanism consists of an arrow pointing towards the target's location and changes state when the target is on-screen. The two states can be seen in Figure 16.

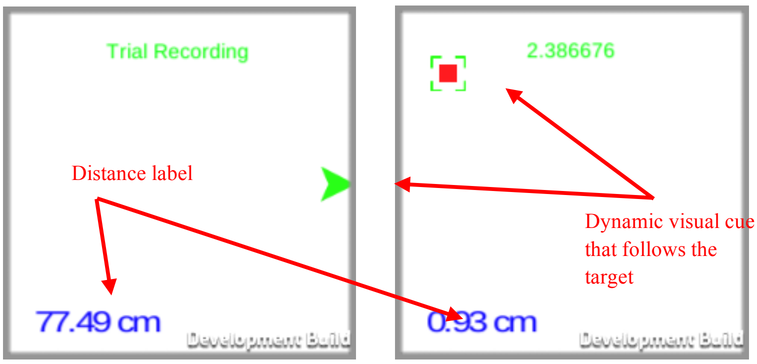


Figure 16 The user interface used for conducting the experimental task.

Apart from having the on-screen indicator, I have created a distance label that updates itself in real-time, thus providing the user with information for the whereabouts of the target. By having this distance counter, the user can adapt his/hers movement speed when getting closer to the screen's center. This improved trial times considerably and minimized overshooting when scrolling during task recordings.[4]

3.12 Logging data

Every movement of each object (smartwatch and finger) is recorded on every frame pass by the laptop. Thus, log files are generated for each trial with all necessary information such as the smartwatch's position and rotation in world space, finger's position in world space, target location relative to the watch, the number of pans that the user had triggered, a *boolean* value that denotes if the finger is in the clutch area and finally the timestamp. All log files contain the following information:

- **user_id** = unique user id for each participant
- **trial_type** = the type of interaction being investigated (mid-air pointing or touch+inertia)
- **target_start_pos** = the initial target position (near, medium and far ranges) relative to the smartwatch
- **sw_pos_x, sw_pos_y, sw_pos_z** = smartwatch's real world position at a specific moment in time
- **sw_rot_x, sw_rot_y, sw_rot_z, sw_rot_w** = smartwatch's rotation at a specific moment in time represented as a quaternion[5]
- **finger_pos_x, finger_pos_y, finger_pos_z** = pointing finger's real world position at a specific moment in time
- **target_pos_x, target_pos_y, target_pos_z** = target's real world position at a specific moment in time
- **target_local_x, target_local_y, target_local_z** = target's local position relative to the smartwatch at a specific moment in time
- **nr_of_clutches** = number of clutches that the participant made since the start of the trial
- **is_panning** = flag that indicates that the user is panning through the interface at a particular moment in time
- **timestamp** = recorded moment in time

These log files will then be analyzed and evaluated upon completing the empirical study. I will comment the results in a later chapter.

3.13 Limitations

During the prototype's development I have stumbled upon different issues and inconveniences that impacted the final results. Some of these issues I have managed to solve, but others could not be fixed. In this subchapter, I will be talking about the problems that have an impact on the system's performance, accuracy and usability and could not be fixed or removed entirely.

[4] This is my subjective opinion without any data to support this statement.
[5] Unit quaternions, also known as versors, provide a convenient mathematical notation for representing orientations and rotations of objects in three dimensions. Compared to Euler angles they are simpler to compose and avoid the problem of gimbal lock. Compared to rotation matrices they are more compact, more numerically stable, and may be more efficient.

Important here to note is that I have tried to come up with different workarounds and solutions to minimize the impact on the final results of the experiments.

3.13.1 Network lag

Because of the system is composed of 3 devices communicating in a daisy-chain structure through the local network (see Figure 11), it comes to no one's surprise that there is a delay between the user's arm movements and the interface updates on the smartwatch. This delay only affects the mid-air pointing task because the positions of the smartwatch and finger must be processed on the laptop and then streamed to the watch. Because this lag could not be avoided, I have tried to reduce the overall overhead by streaming the positions to the watch right after I have finished processing the data frame received from the OptiTrack server. When streaming, the packages that are being sent across the network are as small as possible (20 bytes per payload) on an unreliable channel, which means packages can be dropped and the watch only receives and processes the latest scene state.

3.13.2 Motive accuracy problems

The OptiTrack setup, that I have used, had only 12 functional cameras (out of 24) and severely impacted the performance of my system. The entire system became very sensitive to other objects in the room that might have been mistaken for reflective markers. Because I am tracking the finger's position with only one marker that is attached on the user's fingernail, the probability of that marker to be occluded or mistaken for some other object in the room was very high. The tracking of the smartwatch, on the other hand, was less of a concern. It had been done using a fiber filled nylon plastic rigid base attached to the back of the smartwatch, between the device and the user's wrist. Having 4 markers that represent the rigid body provides more reliable rigid body tracking and even if one or two markers are occluded, the software can predict the watch's position without any problems. I tried minimizing the finger tracking issues with the algorithm that I have presented in an earlier subchapter.

3.13.3 Smartwatch's left and right touch areas

The task that required the user to pan directly on the display (*touch+inertia*) had some imprecise data recordings due to the fact that the watch's operating system reserved the left and right borders for system "call to" actions. Thus, when swiping from outside the watch's border, left or right, the touch motion wouldn't register. When swiping from the bottom or from the top, the event did register and the user could manipulate the interface. This would mean that higher values in completion times should be expected when investigating the directions where the target had spawned initially. I will be investigating this in my analyses in a later chapter.

3.13.4 OptiTrack camera tracking noise

Ideally, the OptiTrack system needs to be recalibrated after each time somebody enters the room. The cameras that are attached on the door-sided wall may lose calibration due to the vibrations generated when someone opens and closes the door. This will move the some of the cameras ever so slightly that will add noise to the system and affect my results when tracking the participants. I tried recalibrating the system as often as possible so I would get higher quality data when conducting my experiments.

Another annoying situation was the fact that I had to instruct some of the participants to remove parts of their clothing, such as shoes that had reflective materials, or sweaters that contained some materials which added a huge amount of noise to the camera system. This is a very important limitation that affects the current setup's usability.

4 DESIGN OF AN EMPIRICAL MID-AIR INTERACTION STUDY

This chapter will describe the design of this thesis' empirical study. I decided to evaluate the interaction model designed in chapter 3 and comparing it with the traditional touch input. The experiment will provide data which I will use to evaluate the performance, accuracy and subjective user opinions of each interaction model.

The first part will describe my design proposals and control methods. Afterwards I will describe my initial hypotheses. The next part will include the reasoning behind the study's design and procedures. Lastly I will talk about the results and findings that will sum up in the next chapter where I will state my conclusion on the project, the strengths and weaknesses, and future improvements and suggestions.

4.1 EVALUATED INTERACTION METHODS

The scope of this thesis is to primarily analyze two main interaction methods: *touch+inertia* and *mid-air pointing*.

4.1.1 TOUCHSCREEN INPUT METHOD

Touchscreen will act as the base mode of interaction. Currently, smartwatches use the same touch screen interaction model as other smart devices (smartphones, tablets, PDAs etc.). This method reveals some challenges that have been described in Chapter 2.1.

A task using the smartwatch's touchscreen consists of interactions that use only the device's touchscreen by manipulating the on-screen space. Thus, the amount of space that the user can move is constrained by the display's touch surface area. Occlusion becomes an important factor here as well, as the user often cannot see what he/she is touching.

As previously stated in an earlier chapter, using a touch interface without inertia only brings discomfort, tiredness and annoyance for the user. The performance impact is high as well when comparing completion times and number of clutches necessary for task completion[6].

4.1.2 OFF-SCREEN MID-AIR POINTING INPUT METHOD

The user manipulates the onscreen display by mid-air pointing using his/her preferred hand which moves the smartwatch's viewport in the desired direction. This is the primary focus of the thesis. Investigating the interaction model and comparing the results with the *base* touchscreen input method. The users must move their preferred hand through their respective range of motion, to move the watch's viewport and complete the predesigned task. A clutch will be simulated by having the hand in a specific clutching area around the display. I will discuss in detail the design of this procedure in the next subchapter.

4.2 HYPOTHESES

In order to determine the ergonomics of the developed prototype, a user experiment was conducted. The goal of the experiment was to understand, if the *mid-air interaction model* is an improvement over the well-established direct *touch input* for smartwatch off-screen space

[6] This is a subjective statement to which I have no data supporting it.

interaction. Furthermore, I wanted to investigate the usability of the prototype measured on different variables. I defined the following hypotheses for this study:

- **Hypothesis 1 (H1):** By having a larger area of input, I expect that the trial times will improve significantly by using the Off-screen air pointing method. Users will be able to complete the same tasks faster by having the ability to interact with a much larger input area.
- **Hypothesis 2 (H2):** Off-screen air pointing will have less number of clutches for completing the task. By giving the user more space for maneuvering the view port, he/she will need less clutch/release interactions.
- **Hypothesis 3 (H3):** Off-screen pointing will be less fatiguing than traditional touch.

4.3 EXPERIMENT SETUP

For my experiments, I used an OptiTrack motion capturing system with 12 OptiTrack Flex3 cameras. Each camera is capable of 100 FPS marker tracking with a maximum resolution of 640x480. Each camera has 26 infrared LEDs and has an average latency of 10ms[7]. As previously mentioned in another chapter, these cameras capture the positions of different markers in the room. The OptiTrack server calculates and streams the smartwatch and the user's finger positions over the local network to a Mid 2015 Retina 15-inch MacBook Pro running Windows 10 (a detailed overview can be seen chapter 3.4). The laptop then receives the raw data, records log files for each trial and submits commands and target positions to the smartwatch, which is also connected to the local network. The watch and laptop have a two way communication stream that synchronizes program state and events (touching, moving the target, starting and ending a trial etc.).

Multiple recalibrations of the OptiTrack system have been done. This ensured that my log files had not been compromised by tracking noise and camera errors.

4.4 PARTICIPANTS

The user study had 8 participants, 6 male and 2 female, with ages ranging between 23 to 34 years old (*Average=26, Median=25*). None of the participants required any optical aid and all of them were right handed (this meant wearing the watch on their left wrist and panning with their right index finger). The subjects were sampled among students, family and friends and all were familiar with a mobile device and owned one as well. Important to point out for my study is that only one participant owned a smartwatch. Only one participant had never used a mid-air interactive system before. All participants have rated themselves as experienced or highly experienced with mobile touch devices and were comfortable in using them.

4.5 OBSERVED VARIABLES

4.5.1 INDEPENDENT VARIABLES

- **Interaction method:** Touch+Inertia, Mid-air.

[7] Values taken from OptiTrack's website (http://optitrack.com/products/flex-3/)

- **Target distance:** Short (*0.2-0.3 meters*), medium (*0.5-0.7 meters*) and long (*1-1.4 meters*) distance.
- **Target Direction:** Initial target's cardinal direction: N, NE, E, SE, S, SW, W, NW.

4.5.2 DEPENDENT VARIABLES

The list of dependent variables that were either monitored during the experiment.

- **Task completion time:** The measured time in milliseconds that elapses between the moment the user starts the a trial, until the moment when the target has been translated in the smartwatch screen's viewport minus the set 3 second dwell time.
- **Number of clutches:** The number of clutches the user has to do in order to complete the task.

4.6 ADDITIONAL RECORDED DATA

- **User satisfaction:** Results of the questionnaires that measure user satisfaction and comments regarding each technique
- **Finger fatigue:** Comparing how cumbersome and how fatiguing each of the interaction methods were.
- **Relative user finger's movement:** Measured by the position of the user's pointing finger relative the smartwatch, during the experiment.
- **Target position:** The position of the target relative to the smartwatch's position.

4.7 TASK DESIGN

I have chosen a targeting acquisition task for my experiments, where the target is spawned randomly, to reduce the risk of having biased results for the later positions, because of the user's ability to adapt, learn and improve his techniques while conducting the experiments. The positions of each target point can be seen in Figure 17, where they are aligned in a grid structure. The positions are grouped in 3 separate distance categories: short (*0.2-0.3 meters*), medium (*0.5-0.7 meters*) and long (*1-1.4 meters*).

Participants are requested to complete a docking task by having to locate and drag a target to the center of the smartwatch's display. The surface area where the target is located is numbered in a rectangular grid. The grid is represented by a XY coordinate system with each axis having a range from – 1 to +1 meters. The entire system has been calibrated to work in real world measurements, for example, the distance from the target to the center of the smartwatch corresponds to the exact distance and position in real world space. An illustration of the aforementioned layout can be seen in Figure 17.

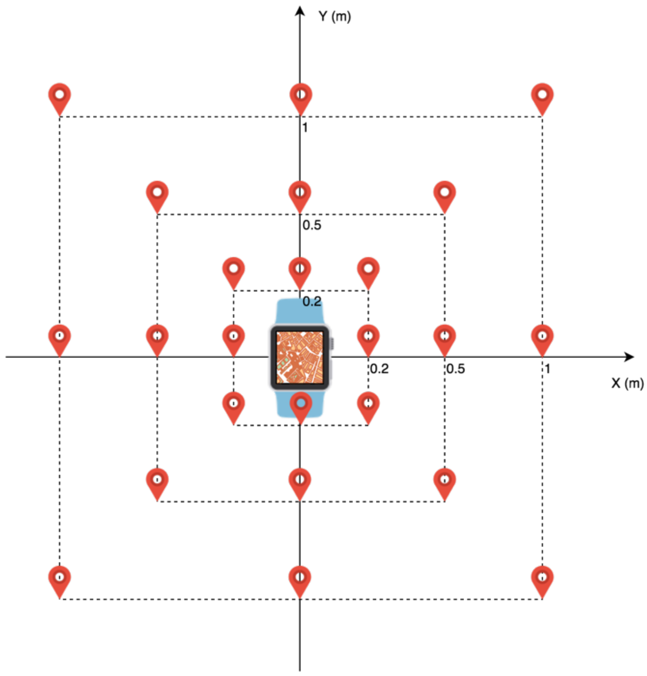


Figure 17 Experiment design that consists of a target acquisition task with points that spawn randomly at predefined positions.

At the start of each task the user's initial position is reset at *(0,0)* and the range observed on the screen is between *-0.02 and 0.02 meter*s on both axes. When transitioning to the next sub-task, the participant is required to move their pointing finger to the starting *(0,0)* position. Only after this step has been completed, the next sub-task will commence.

On the smartwatch's display, there will be a number notifying the participants what position they should dock in the viewport's center. After the target has been centered on screen, there will be a small dwell time of 3 seconds to confirm task completion. The measurements captured from the trials consist of interaction time - from the user's fingertip position located at (0,0) until the docking has been completed (the 3 second dwell time is excluded) - finger and device positions during all trials, as well as the number clutch area entries/onscreen touches.

4.7.1 Touch Interaction Performance

I considered direct touch input as the baseline interface for this study, due to how ubiquitous these interfaces have become in recent years. Most touch interfaces incorporate inertia for speeding up the interaction with elements beyond the viewport area. I decided to enable inertia for the study, mostly because users have become very dependent on it and the smartwatch's touch area was way too small to carry out the task in a reasonable amount of time without it (one of the tasks took 27 min to complete).

4.7.2 Off-Screen Mid-Air Interaction Performance

For the mid-air interaction method, clutching is performed by having a predefined area around the display. This represents the direct world representation of the virtual space of the "*Clutching Area*" modeled in the setup's implementation detail in Chapter 3.10. When the finger is entering this volume space, it triggers a simulated touch that will translate into a drag event for manipulating the virtual world space.

The participant has to follow the on-screen visual cues represented by an arrow that is located at the screen's border. When the target is in movement, the arrow changes position accordingly and transforms itself into a different on screen cue when the target is in view, as shown in Figure 16.

4.7.3 Providing consistency between interaction methods

In order to get an accurate comparison between the two interaction methods I decided to use the exact same scene for all experiments. By having an identical scene across both interaction methods, I can compare recorded values directly. In both instances the data log files have been recorded with the same structure presented in chapter 3.12. The targets spawn randomly in the same predefined positions for both scenes.

During the touch+inertia experiments, the finger's position is tracked by the same motion cameras as in the mid-air experiments. They provide excellent submillimeter precision and would not add any incorrect values to my data.

4.8 Study Design

Each participant will be using and evaluating both techniques. A latin square design has been used to counterbalance the effect that the order of the technique might have on the entire study. This resulted in two sequences of methods that are presented in the table below.

Sequence 1	Touch+Inertia	Mid-air
Sequence 2	Mid-air	Touch+Inertia

Table 1 Latin square for method sequencing.

For each different target position I generate a distinct log file. I went for this approach in case something went wrong during trial recordings, so I wouldn't have to redo the entire trial. Each participant will generate a total of 48 log files, of which 24 are for *touch+inertia* trials and the rest for the *mid-air* interaction trial.

The participants were asked to fill out a questionnaire after each session in order to evaluate the user satisfaction for the individual interaction methods. My questionnaire was made using questions from the standard ISO-9241-9[46] questionnaire that can be found in the Appendix A. The questions gather information about the users subjective experience with the method concerning usability, ease of learning. fatigue and overall satisfaction. In the end the participants would fill out a final question, where he/she is asked to rank the four methods in order of his personal preference.

4.9 Task procedure

The participants received an introduction upon arrival and were asked to pay close attention to the author (myself) when explaining the different aspects of the system. These introductions were performed in private so the other participants would not improve their knowledge on how the interaction method works. The users were encouraged to speak out about any comments or misunderstandings they might have had. I then proceeded to remind the participants that they were not being tested, but that it was the methods that I evaluated, thus relieving some of the stress of performing well under observation.

Before the study could start, they received a training session by the author, that explained how one would interact with the smartwatch and what the task consists of. Before strapping the smartwatch with the markers behind it onto the wrists of the participants, I first carried out a demonstration on how I would control the user interface under different circumstances. Because I was using an OptiTrack system with only 12 cameras, the risk of marker swapping increased considerably. I explained that when trying to pan through the interface, the users should be careful when gliding the pointing finger marker close to the markers attached to the smartwatch. Also I had cautioned the participants to not conceal the other markers with their hands while panning close to the smartwatch. I also advised them to pay attention to the watch's user interface, especially in the mid-air pointing trial, because their movements might not register fast enough due to network lag or load.

After completing my demonstration, I transferred the smartwatch to the participant and instructed him/her to try both interaction techniques. Each user was instructed to enter a map demo scene shown in Figure 18. This part is used for getting people to understand the real-world usage for getting the feel of interacting beyond the display. The user were encouraged to play around with the interface until they felt confident enough that they would want to start the actual trials, which would then be recorded.

Figure 18 Map demo scene where users can test out the mid-air pointing interface that could be used in a real-world usage scenario.

Before beginning any trials, the users had to fill out a part of the questionnaire regarding their age, dominant handiness, whether they had any previous experience with mid-air pointing technologies etc. The questions can be examined in Appendix A.

After the participant had finished all trials covering one of the input methods, he/she was asked to fill out the questionnaire related to the method that they had just completed, before moving onto the next technique. The next method started as soon as the participant was ready to continue, which meant that he/she had the opportunity to take a small break, if he/she desired to do so.

Once all task were completed, the final entries in the questionnaire would be filled out and the participant was asked to give additional feedback about their experience with the methods and rate the two interaction methods.

5 STUDY ANALYSIS AND EVALUATION

In this chapter I will be evaluating and analyzing the data that I have recorded during the user study. I will try to evaluate the hypotheses that I have stated in the previous chapter and report all my findings. In the following subchapters, I will be discussing in detail the implications of my study.

The evaluation consists of two main parts. In the first part I conduct several statistical analyses on the recorded data in order to support or refute my hypotheses. The second part consists of analyzing the filled in questionnaires and the different comments that the participants had during the user study.

I end with my conclusions and additional feedback reported by the participants which can be used for future development.

5.1 DATA CONVERSION

For performing my statistical analyses on the logged data, I had to transform and categorize my data into a separate dataset. I used a python script that can be found in a the archive that has been uploaded along with this document. When analyzing completion times and the number of interactions I decided that I would need the following dataset structure:

- **user_id** = the participant's unique identifier
- **trial_type** = the type of interaction being investigated (mid-air pointing or touch+inertia)
- **target_distance** = initial target positions are categorized into 3 different areas (short, medium and long range)
- **direction_axis** = in which direction is the target initially located (N, NE, E, SE etc.)
- **nr_of_clutches** = the total number of clutches for each trial
- **completion_time** = total completion time calculated by using the following equation: $T_c = t_l - t_i - t_d$ where t_l is the last timestamp in the log file, t_i is the first timestamp and t_d is the dwell time of 3 seconds.
- **task_number** = the order in which the random trials have been conducted (this will be relevant for investigating learning effects)

5.2 ANALYSIS OF NUMBER OF INTERACTIONS

An important factor in measuring the performance of interaction methods is the number of interactions a user has to do to complete a predefined task. I define an interaction as a touch and drag on the smartwatch's touchscreen for the *touch+inertia* interaction method and as a hover and drag when the user is in the clutching area for the *mid-air* interaction method. I define the term *clutch* a transition between interaction states (for example *idle* and *dragging*). I use the terms clutch and interaction equivalently.

5.2.1 DATA TRANSFORMATION

Because the number of clutches differ greatly between interaction methods, my data was heavily right skewed, as shown in the scatterplot and histogram in Figure 19. The logarithmic transformation often equalizes variances for data in which the mean and the variance are positively correlated[47]. A positive correlation means that groups with large averages will also have large variances (in ANOVA). Before applying the log function to my data points, I had also added a value of 0.5 to each data point because when taking into account the most of the values specifically for the mid-air interaction method would have values of 1 and $\log(1) = 0$, thus it would affect my variance analysis. This approach was recommended by Sokal and Rohlf (1995)[48]. As we can see in the figures below, by applying $Y^* = \log(Y + 0.5)$ transformation on the number of clutches I could continue in finding our model.

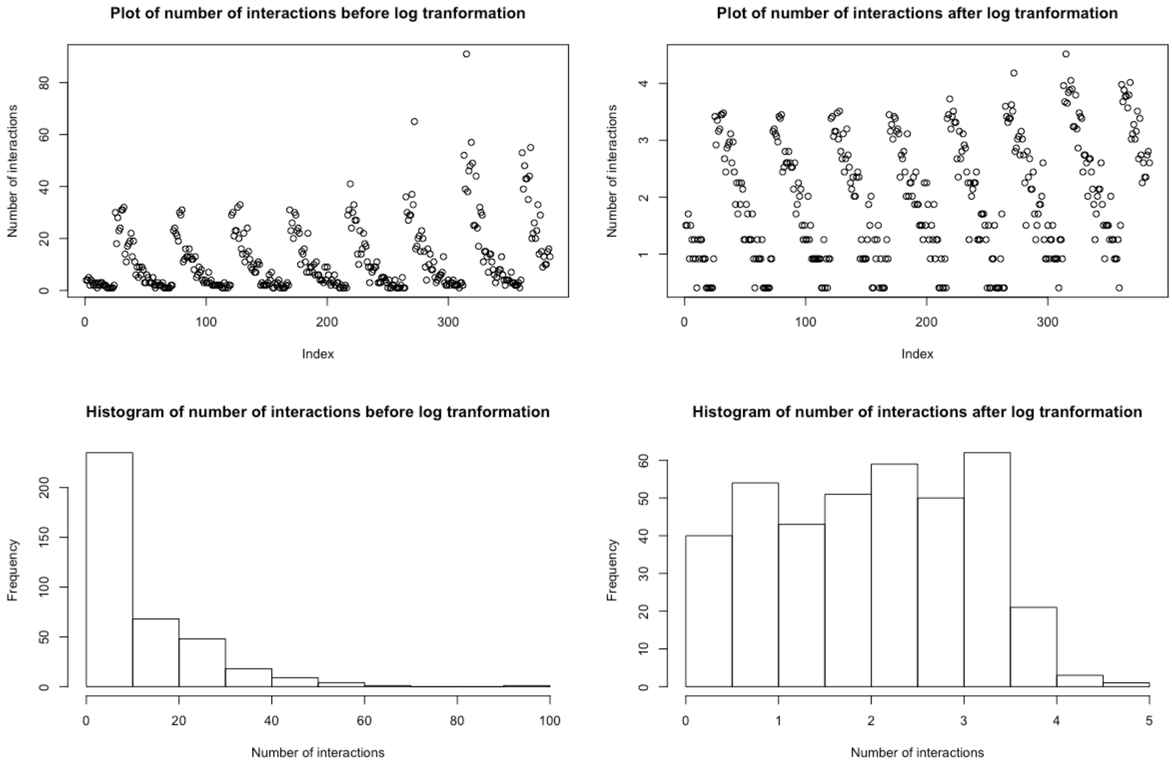


Figure 19 Scatterplots and histograms of the number of clutches before and after aplying the log transformation.

5.2.2 FINDING THE RIGHT MODEL

Because I expected more variables than just the interaction method to have an effect on the number of clutches, I included all these in my initial model (Equation 1). In order to identify the right model I used the *backwards elimination method* which means starting out with all candidate variables, and testing if the removal of each variable using a chosen model fit criterion. The main idea is to delete step-by-step the variables that do not provide a statistically significant loss of fit to the model.

Firstly, I am interested in finding if there is a statistically significant difference between a model with an interaction, in which the two independent variables (in my case the interaction method and the target distance) are "crossed" with one another so that there are observations at every combination of levels of the two independent variables, and a model that does not take into account the interaction. The initial formula describing both the fixed-effects and random effects part of the model with can be seen in Equation 1.

$$Y_{model4} = trial\_type \times target\_distance + trial\_type \times direction\_axis + task\_number + user\_id_{random\ effect}$$

Equation 1 Starting model with interaction for number of interactions

Because I included the *task_number,* which is a continuous variable, I performed ANCOVA (analysis of covariance). The hypothesis is that the covariate also contributes to variation in the response variable. When comparing the starting model with the model without the interaction between interaction method and *direction_axis*, described by Equation 2, the models were not significantly different from each other, meaning that the interaction was not significant (p= 0.051) and we can exclude it.

$$Y_{model3} = trial\_type \times target\_distance + direction\_axis + task\_number + user\_id_{random\ effect}$$

Equation 2 Model without trial_type and direction_axis interaction for number of interactions

When comparing the model with interaction between interaction method and *target_distance* (Equation 2) with a model without (Equation 3) the models were significantly different ($p<0.001$).

$$Y_{model2} = trial\_type + target\_distance + direction\_axis + task\_number + user\_id_{random\ effect}$$

Equation 3 Model without interaction for number of interactions

The next step was to reduce the model even further by testing the impact of the covariate *task_number*. By removing the covariate, the initial model was now tested against an ANOVA model (Equation 4) and it was shown that the *task_number* is not significant ($p=0.422$). This would also suggest that there might not be any learning effect involved. I further tested if the target's direction had any effect. After testing against another ANOVA model (Equation 5), it was shown that there is a significant difference ($p<0.001$) between the model that takes the direction factor into account and the one that doesn't. At this point we have found our final model which is defined by the Equation 4 below.

$$Y_{final} = trial\_type \times target\_distance + direction\_axis + user\_id_{random\ effect}$$

Equation 4 Final model for number of interactions

$$Y_{model1} = trial\_type \times target\_distance + user\_id_{random\ effect}$$

Equation 5 Model without direction axis

5.2.3 MODEL VALIDATION

ANOVA models are based on the assumptions that the residuals are normally distributed. Therefore, after I have defined my model, I have tried to validate it by investigating the variance homogeneity of the model's residuals, and the normal distribution of the model's residuals with histogram plot and finally creating a QQ-plot, or quantile-quantile plot, which compares the sample quantiles to those of the normal distribution. As we can see in the figures below, the residuals are normally distributed after the transformation and this implies that our model has the best fit.

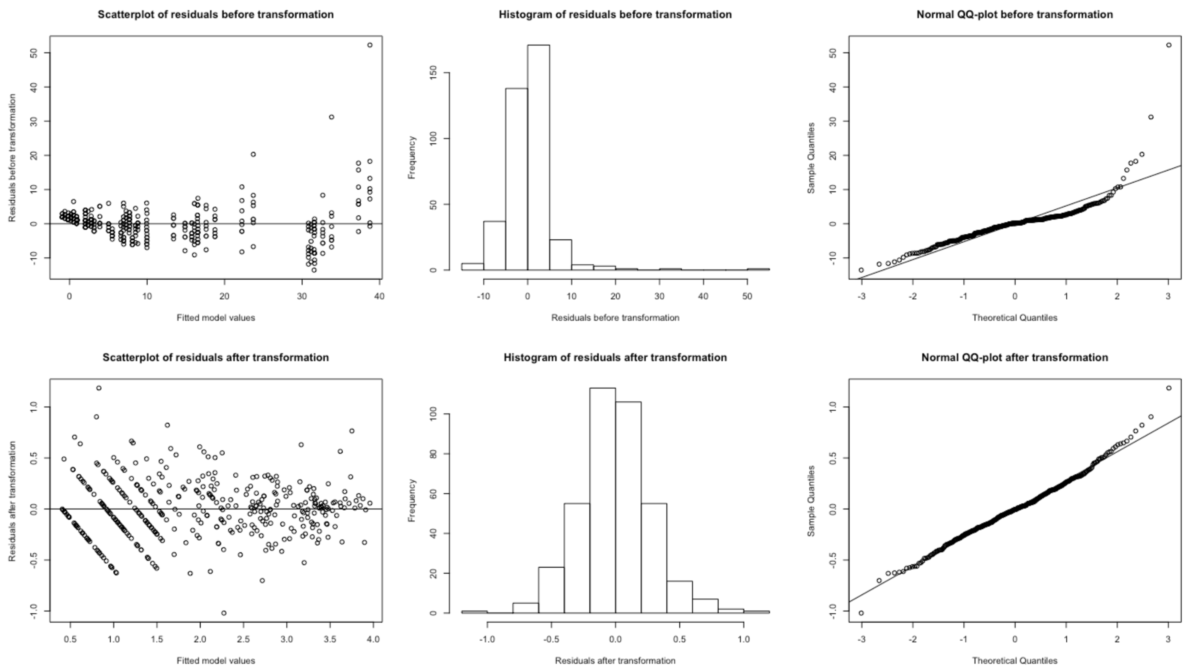


Figure 20 Scatterplot and histogram of the model's residuals, as well as the model's QQ-plot.

5.2.4 INTERPRETING THE RESULTS

There was a statistical significant difference in number of clutches between *interaction method* ($p<0.001$) and *target distance* ($p<0.001$). The significant effect of *target distance* was unsurprising, as further targets require the use of more dragging operations. In Figure 21 we can clearly see that there is a difference between interaction methods ($p<0.001$), thus **Hypothesis 2 (H2),** which stated that *off-screen mid-air pointing* will have less number of clutches for completing the task than *touch+inertia,* is **true**. By giving the user more space for maneuvering the view port, he/she will need less clutch/release interactions. Overall, *touch+inertia* required most operations(μ_{short}=9.131, μ_{medium}=17.401, μ_{long}=31.655) when comparing with the mid-air pointing method(μ_{short}=1.547, μ_{medium}=2.563, μ_{long}= 4.230).

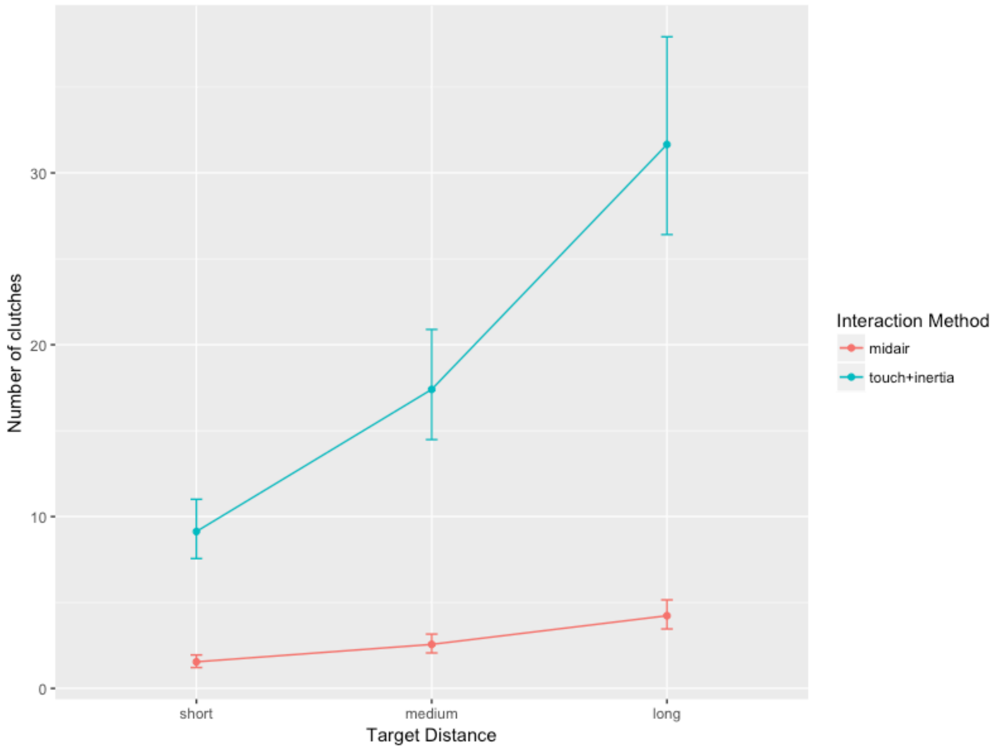


Figure 21 Number of clutches against distance grouped by interaction method

When stratifying the model according to the targets initial direction we can see that the directions have the same order on the graph for the *touch+inertia* and *mid-air* interaction methods. That is because no statistical significant difference was found when testing for an interaction between the direction and interaction method. Some of the directions were significantly different from each other, and I have included the *p-values* in Appendix C.1. I did not perform a detail analysis here, because I considered it to be out of the current scope of this thesis.

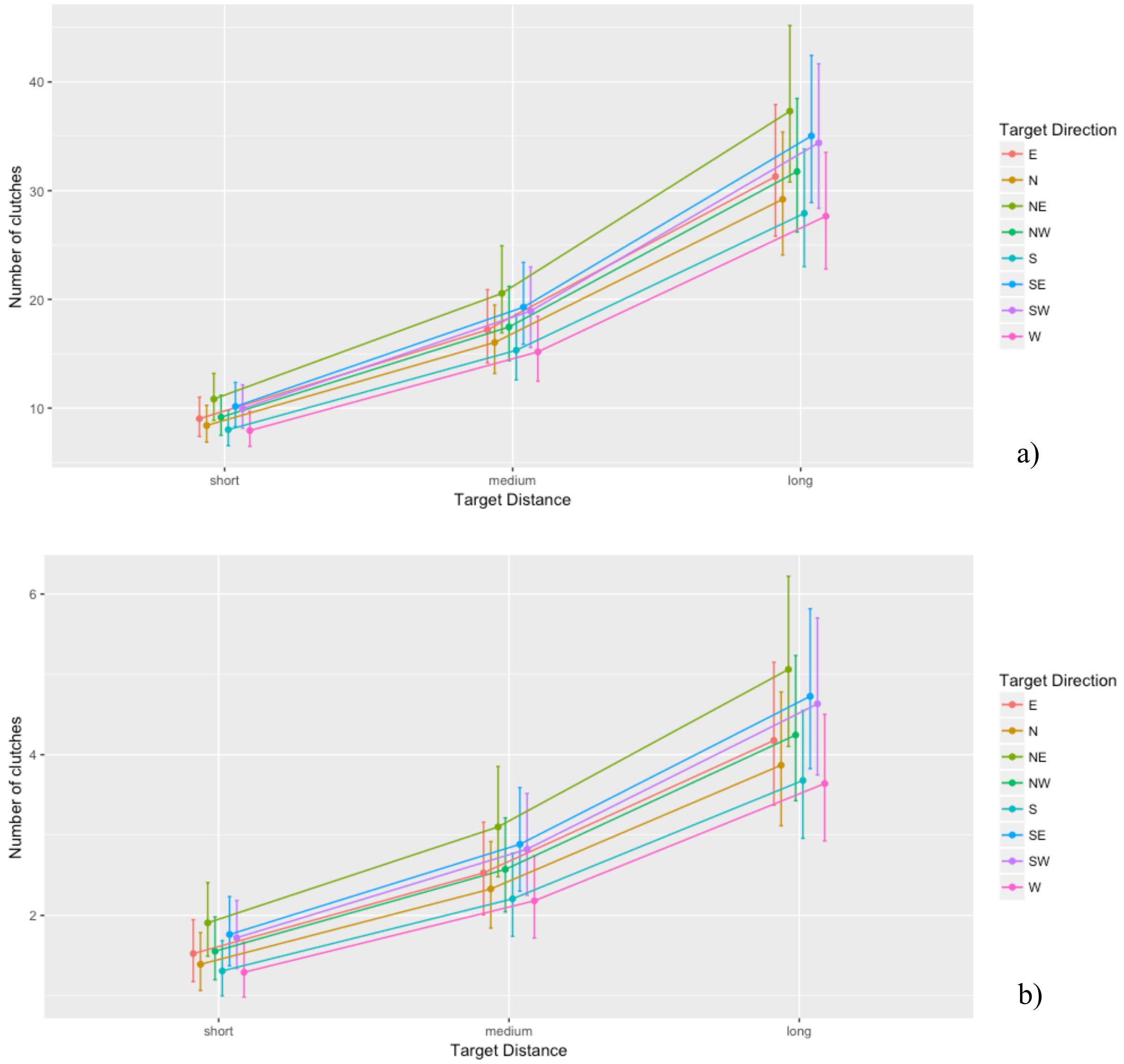


Figure 22 Number of clutches against distance stratified by direction for a) touch+inertia and b) mid-air interaction method

5.3 Analysis of Task Completion Time

Another important factor in measuring the performance of interaction methods is the analysis of *completion time.* This means how much time does the user spend for finishing a predefined task. I define the completion time for each task as the time elapsed starting from the moment the user starts interacting with the smartwatch to the moment the trial ends by docking the target in the center of the screen.

5.3.1 Data Transformation

As we can see from Figure 23, the data for completion times is slightly right skewed. In order to make my data normally distributed, I have used the power transformation $Y^* = 1/\sqrt{Y}$, which seemed to provide the best results.

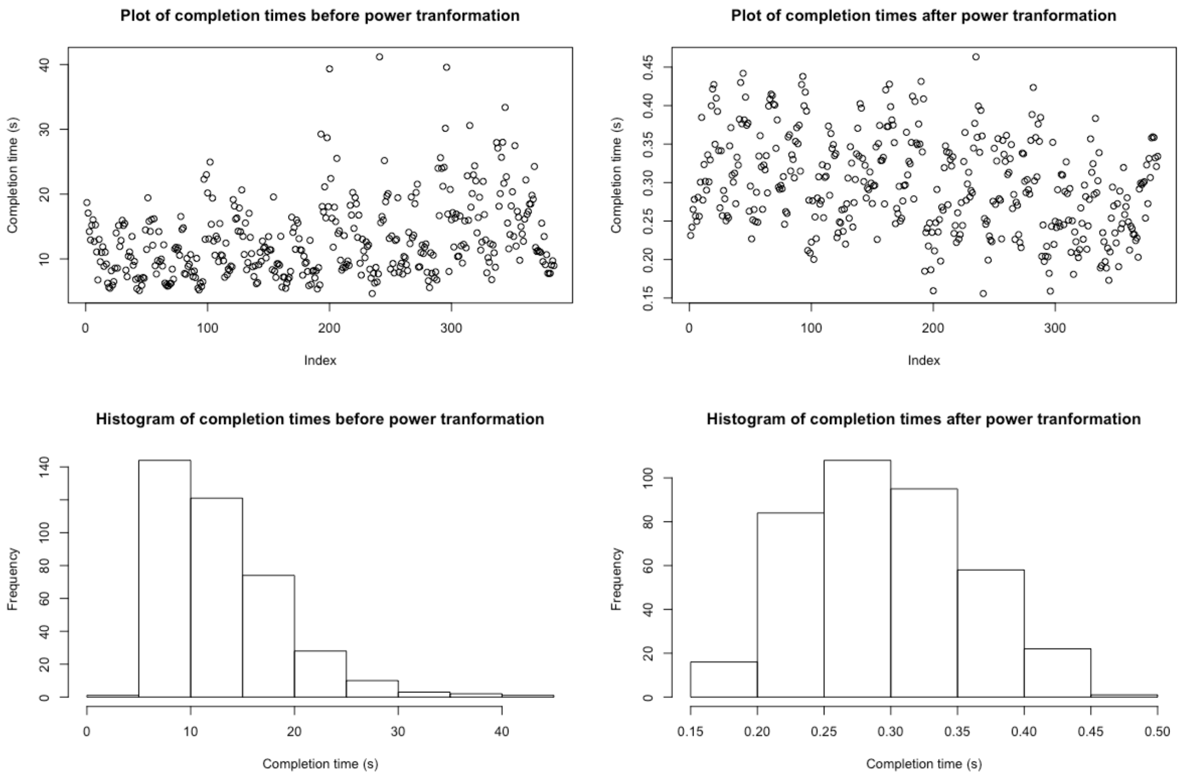


Figure 23 Scatterplots and histograms of the completion times before and after applying the power transformation.

5.3.2 Finding the right model

For finding the model for task completion time, I used the same approach as the one used in establishing the model for the number of interactions. I started out by including all variables in my initial model. The initial formula describing both the fixed-effects and random effects part of the model with can be seen in Equation 6.

$$Y_{model4} = trial\_type \times target\_distance + trial\_type \times direction\_axis + \ task\_number + \ user\_id_{random\ effect}$$

Equation 6 Starting model with interactions for completion time

When comparing the starting model with the model without the interaction between *interaction method* and *direction*, described by Equation 7, the models were not significantly different from each other, meaning that the interaction was not significant (*p=0.397*) and we can exclude it. Then I checked if there is an interaction between the *trial type* and *target distance*.

$$Y_{model3} = trial\_type \times target\_distance + direction\_axis + \ task\_number + \ user\_id_{random\ effect}$$

Equation 7 Model without trial_type and direction_axis interaction for completion time

The *task_number is included,* therefore I performed ANCOVA (analysis of covariance) tests on the completion times. When comparing the model, described by Equation 7, with the model without the interaction, described by Equation 8, the models were not significantly different from each other (*p=0.448*), meaning that the interaction was not significant and we can exclude this one as well.

$$Y_{model2} = trial\_type + target\_distance + direction\_axis + task\_number + user\_id_{random\ effect}$$

Equation 8 Model without interaction for completion time

I reduced the model even further by testing the impact of the covariate *task_number*. By removing the covariate, I now perform an ANOVA test and it has shown that the *task_number* is not significant (*p=0.521*) for the second model. This would also suggest that there might not be any learning effect involved. I further tested if the target's direction had any effect. After testing against another ANOVA model, it had shown that there is a significant difference (*p<0.001*) between the model that takes the direction factor into account and the one that doesn't. At this point we have found our final model which is defined by the equation below.

$$Y_{final} = trial\_type + target\_distance + direction\_axis + user\_id_{random\ effect}$$

Equation 9 Final model for completion time

5.3.3 MODEL VALIDATION

I validated my model using the same steps as before, by testing the variance homogeneity of the model's residuals, testing the normal distribution of the model's residuals with histogram plot and finally creating a QQ-plot, or quantile-quantile plot, which compares the sample quantiles to those of the normal distribution. As we can see in the figures below, the residuals are better distributed after the transformation and this implies that our model has a better fit.

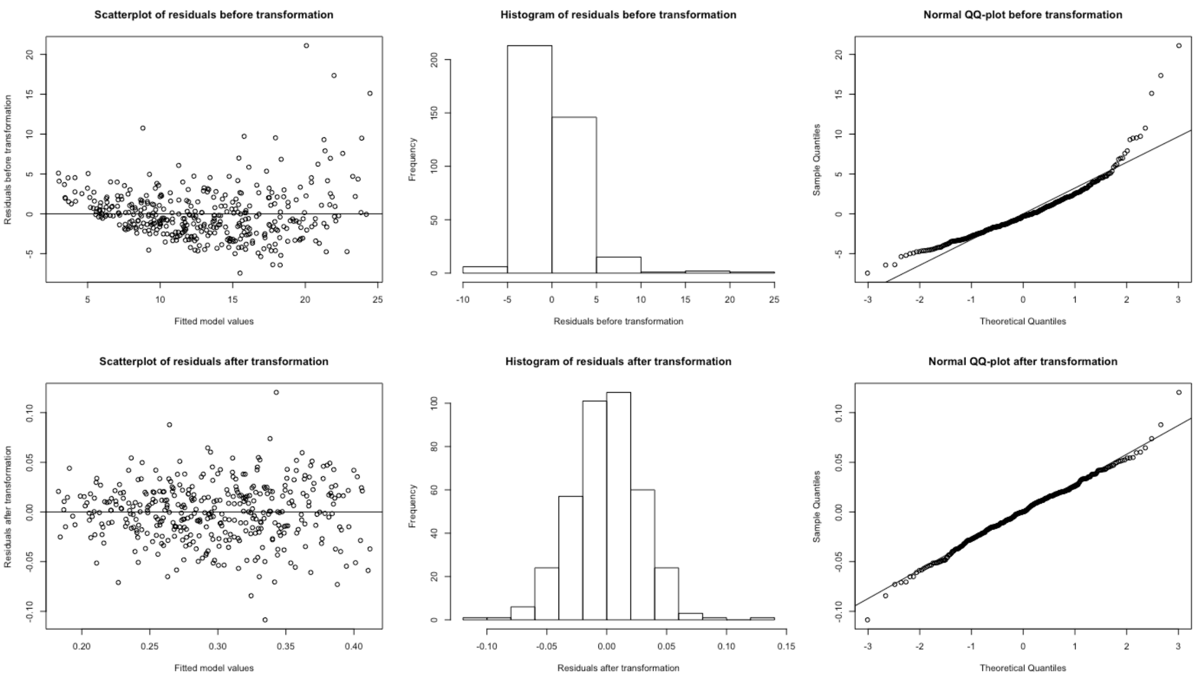


Figure 24 Scatterplot and histogram of the model's residuals, as well as the model's QQ-plot.

5.3.4 INTERPRETING THE RESULTS

There was a statistical significant difference in completion times between *interaction method (p<0.001)* and *target distance (p<0.001)*. Pairwise comparisons revealed that *touch+inertia* was faster than *off-screen mid-air pointing* interaction method, which means that **Hypothesis 1 (H1)**, that the trial times will improve by using the off-screen mid-air pointing method, is

false. In Figure 25 we can see the completion time measured in seconds against target distance stratified for interaction method. There is a difference between interaction methods (*$p<0.001$*). Users completed the same tasks faster by using the *touch+inertia* method. Overall, *touch+inertia* was faster(μ_{short}=7.620, μ_{medium}=10.669, μ_{long}=15.661) than *mid-air pointing.* (μ_{short}=8.518, μ_{medium}=12.181, μ_{long}=18.409)

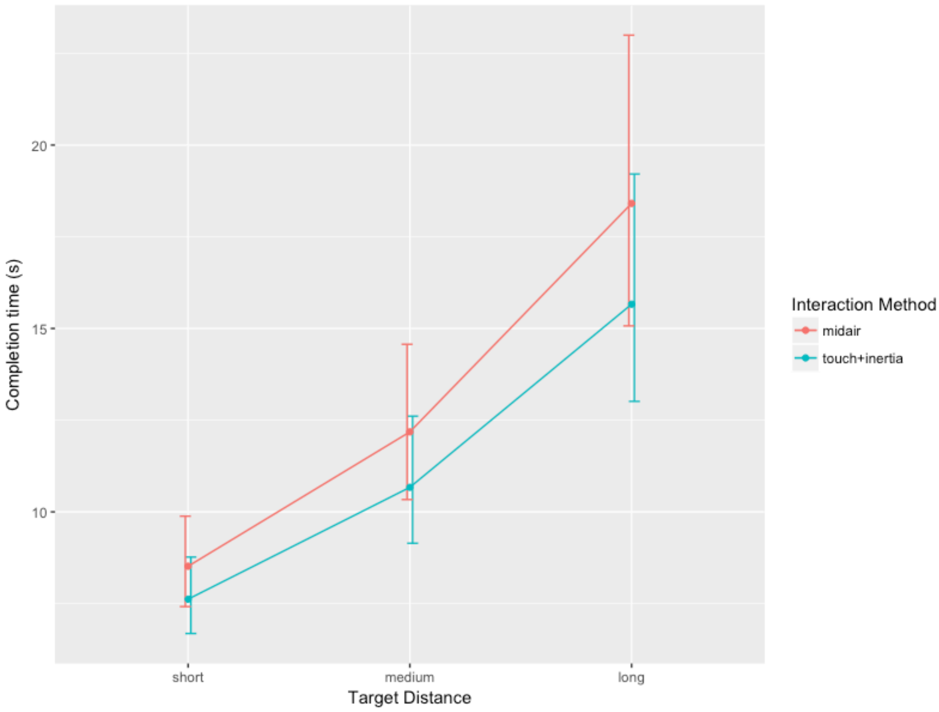


Figure 25 Completion time (s) against target distance stratified for interaction method.

When stratifying the model according to the targets initial direction we can see that the directions have the same order on the graph for the *touch+inertia* and *mid-air* interaction methods. That is because no statistical significant difference has been found when testing the interaction between the direction and interaction method. Some of the directions were significant from each other, but I decided not to include them in the analysis. However, I have included the *p-values* in Appendix C.2.

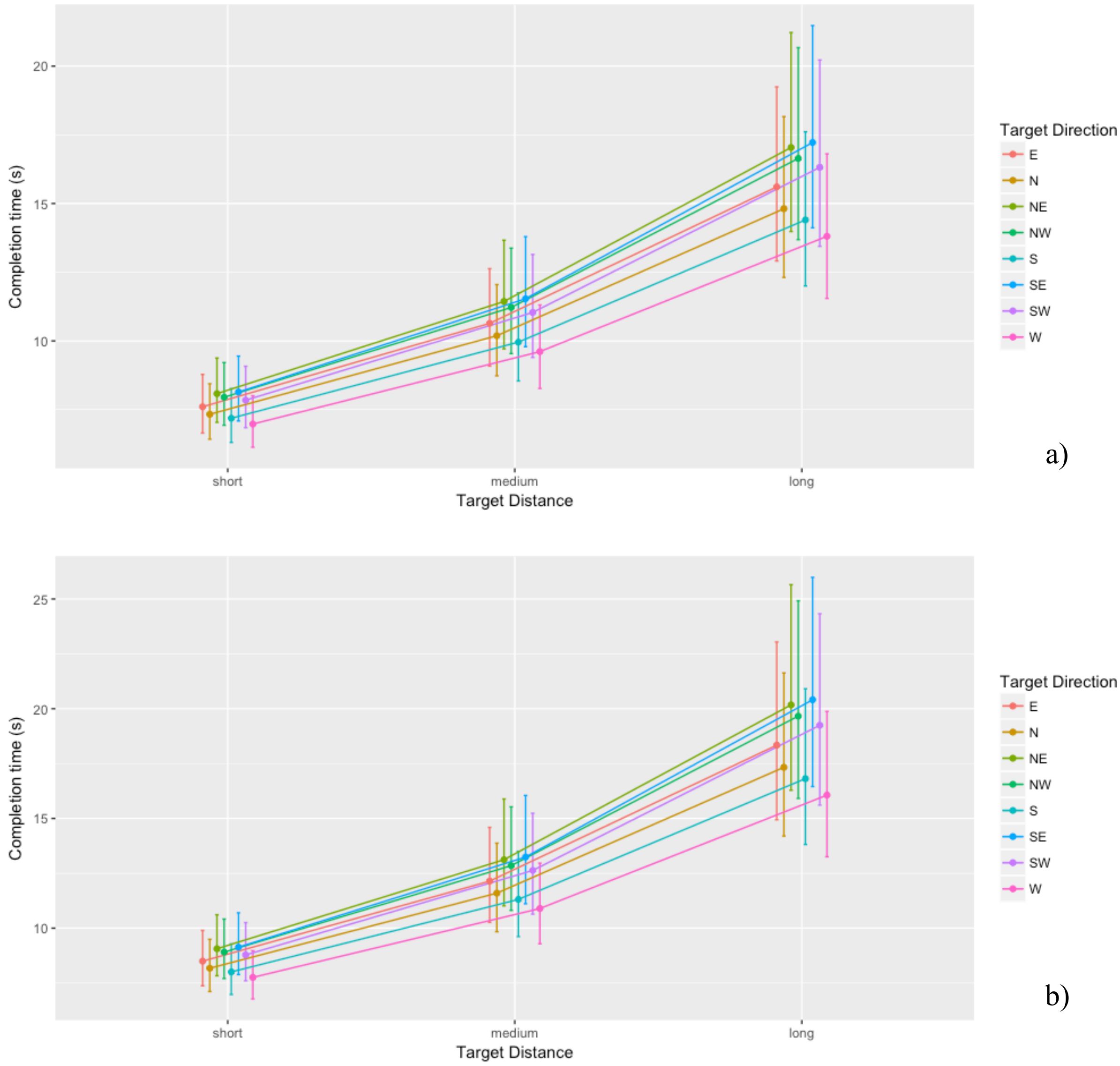


Figure 26 Completion time (s) against distance stratified by direction for the a) touch+inertia and b) mid-air interaction method

5.4 QUESTIONNAIRE ANALYSIS

The participants were asked to fill out a questionnaire after finishing both interaction experiments. In the last part of the questionnaire the users had to give an overall subjective ranking on which interaction method was better. After collecting the results, 62.5% preferred *touch+inertia* over the *md-air* pointing technique. When looking through the feedback that the users provided, I noticed that multiple people have stated that it felt that the touch interface was more responsive and more comfortable due to the fact that the interaction starts only when physically touching the display, whether as in the mid-air technique, moving your arm by mistake would trigger unintentional interactions on the smartwatch. For each technique, a set of questions have been selected to give an idea on how each input method performed, in terms of user satisfaction and ease of use. An overview of the questions can be seen in the table below as well as in the Appendix A.

Abbreviation	Question	Rating
Q1	Smoothness during operation	1 (Very rough) – 5 (Very smooth)
Q2	Operation speed	1 (Too fast) – 5 (Too slow)
Q3	Finger fatigue	1 (None) – 5 (Very high)
Q4	Wrist fatigue	1 (None) – 5 (Very high)
Q5	General comfort	1 (Very uncomfortable) – 5 (Very comfortable)
Q6	Overall, the input method was	1 (Very difficult to use) – 5 (Very easy to use)

Table 2 Question summary

In the figure below I have plotted the mean ratings for each question. To analyze the data I have used the Pearson's chi-squared test (X^2) which is a statistical test applied to sets of categorical data (in my case the rating from 1 to 5) to evaluate how likely it is that any observed difference between the sets arose by chance.

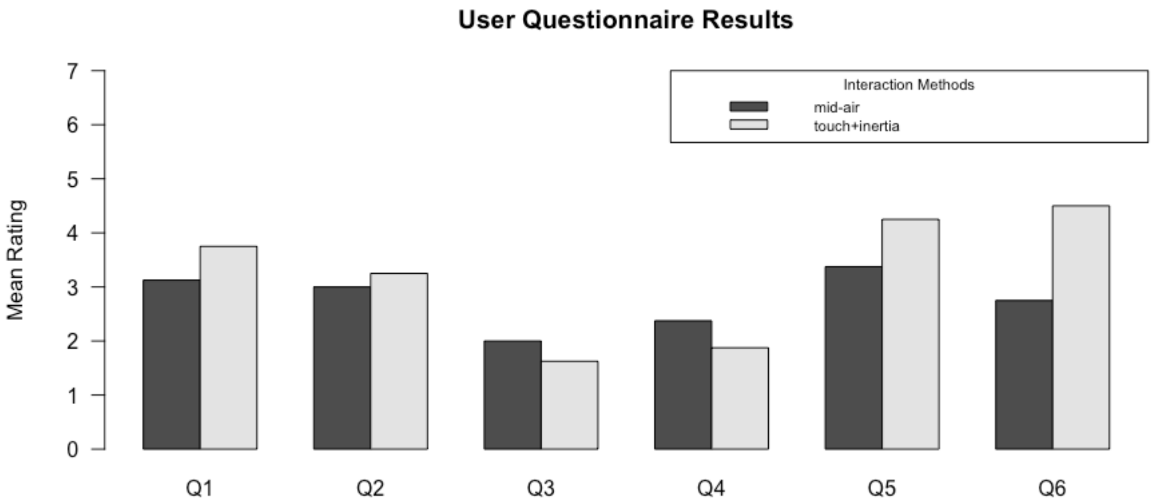


Figure 27 Questions 1 through 6 mean rankings plotted by interaction technique.

My tests show that for questions Q1($p=0.064$), Q2($p=0.513$), Q3($p=0.766$), Q4($p=0.391$) and Q5($p=0.276$) there is no statically significant difference between the two interaction methods. Question Q6($p=0.021$) is the only one that showed a statistical significant difference in favor of the *touch+inertia* interaction method. Even though the mean ratings for questions Q3($\mu_{mid\text{-}air}=2$, $\mu_{touch+inertia}=1.625$) and Q4($\mu_{mid\text{-}air}=2.375$, $\mu_{touch+inertia}=1.875$) are favorable towards the off-screen interaction method, the **Hypothesis 3 (H3)**, which states that off-screen mid-air pointing will be less fatiguing than the traditional touch input, is **rejected**. Nevertheless, my data suggests that *touch+inertia* performed better in regards to the overall ease of use compared to *mid-air* pointing. Overall, *touch+inertia* performed better than *mid-air* pointing across the board.

6 DISCUSSION & FUTURE WORK

In Chapter 4.2, I have hypothesized that, by having a larger area of input, the trial times will improve significantly by using the off-screen mid-air pointing method compared to touch. From the results above, I have to conclude that it performed worse, and in some cases on par, than *touch+inertia*, which means that I have **rejected hypothesis H1**. For **Hypothesis H2**, my results confirmed that off-screen mid-air pointing had significantly less number of clutches for completing the task, as a consequence of this the hypothesis has been **supported**. I believe that by giving the user more space for maneuvering the view port, he/she needs less clutch/release interactions. When analyzing the user survey, I have found that there was no difference between the two interaction methods except when investigating the overall ease of use and fatigue ratings. Therefore, I have to **reject hypothesis H3**, because off-screen pointing performed similarly to touch when I analyzed the users' fatigue ratings. One observation that has been made during the user study by the participants was that the they were feeling more tired in the shoulder region when using the *mid-air pointing* interaction method.

During the *mid-air pointing* experiments, participants often adjusted their speed of panning when closing in on the watch's center. Some suggested that the panning should be "harder" when getting close to the center. This would have been easily implemented by adding a parameter that would adjust the scrolling, but I considered that the development of this feature would affect the results of what I had wanted to investigate. The purpose of my thesis was to investigate how people would interact off-screen on a 1-to-1 scale and without the need of extra features that would modify the user's natural arm movement.

Also for the *mid-air* method, when interacting with the interface, one participant discovered that he could move both his arms towards each other so he can perform the task faster. Although his approach did work, he did say that he felt a little bit uncomfortable doing so.

6.1.1 ADDITIONAL USER FEEDBACK

During the study, I collected supplementary comments from the participants and discovered some very interesting insights. Firstly I will tackle the advantages and disadvantages that were reported for the touch+inertia input method. Four out of eight participants reported that *touch+inertia* was very intuitive and required no learning in using it; this is due to the fact that they were already very experienced with mobile touch devices. Three users noted that the *touch+inertia* interface was more responsive and had significantly less input lag compared to *mid-air pointing*. Some users also stated that the *touch+inertia* interaction method was more precise in completing the experimental task. Drawbacks were also reported for the current implementation of the *touch+inertia* interface, however. Three participants noted that the inertia wasn't additive (similar to list scrolling inertia) and would then not behave as they've expected it to. Two users reported that they could not see the target at times, because of their finger occluding the display and also the small screen limited their ability to scroll.

Switching over to the *mid-air pointing* method, most users have reported that it was difficult to "guess" the clutching area. They had a hard time entering and leaving the interaction area, thus making it hard to control and complete their target acquisition task. One participant stated that there was problem with her hoody. She stated that she "had to take it off because it was interfering with the cameras". Another user reported that he found himself covering some of the screen with the pointing finger, when he tried to keep the marker within the screen. Other noticed the lack of other guidance queues, other than the visual one on screen. However, participants did report some advantages in using the *mid-air pointing* interaction method. Three users noted that it felt very useful for targets located at a far distance. Two persons reported that it was intuitive to use, although they "wouldn't use it on a bus" due to social awkwardness and privacy issues. One user was very happy to say that he learned to use both hand to interact faster, by moving the wrist with the smartwatch with the pointing finger towards each other in the same time.

6.1.2 Possible sources of error

Even though I have tried to minimize sources of error, I have made some mistakes when conducting the experiments. The most important, in my opinion, was the fact that I have not conducted repeated measurements per participant. By doing so I am unable to see if there is any learning involved for each user that could have a significant impact on completion times. That being said, I also consider that by keeping the trials short (an average user would finish all tasks in about 25-35 min) the participants felt less fatigued and would be more objective when answering the questionnaire.

As I have previously mentioned in Chapter 3.13, the prototype had some technical limitations that could impact my results. Taking into account all those impediments, it is clear that further work is encouraged.

Another source of error could be the participants themselves, because most of them are composed of family and friends and could give biased results. However, the statistical analysis has been done on the recorded trial data which cannot be biased.

6.2 Future work

The prototype presented in this thesis was designed to investigate how users would interact with a smartwatch through mid-air pointing. The research that had been carried out elucidated how well this interaction method affected task performance and user fatigue when performing a target acquisition task. As discussed earlier, the prototype isn't perfect and I am proposing some areas of focus for future research: Improving the tracker, investigating the usefulness for different tasks, discovering other techniques for handling off-screen space for smartwatches.

6.2.1 Improving the prototype

As mentioned in a previous chapter, there were some problems in marker tracking. This would have been easily resolved by enabling the other 12 cameras in the room. During the experiments I had captured additional data that I haven't used in my analysis. The log files contain positional data for the finger, smartwatch and target for each participant during the entire experiment. This data could provide some additional insights for future research into what off-screen area do people use the most.

The current prototype only works in a room with motion cameras and having reflective markers attached to the finger and the device. This approach decreases ease of use considerably and constrains the user to only use the system in an enclosed and controlled environment. As we have seen with one participant, the issue of reflective garments can disrupt the way the prototype works. Future work should focus on exchanging the current tracking technology with one that can perform equally or better than the ones presented in Chapter 2.4 when discussing related work.

Future research should also focus on eliminating input lag, because the transition between the idle and drag is noticeable and is affecting the users' task performance. By decreasing the input lag, users can focus more on the task and will not feel the frustration of adjusting their arm position.

Another recommendation for future research is to find the best interaction shape for triggering the pan gesture. The current version extends the 2D plane but the users have a hard time understanding and visualizing where the plane is. Therefore the users need to spend additional time in adjusting their finger position in order to complete the task.

6.2.2 Usefulness on different tasks

The prototype has been tested only for a target acquisition task, for example finding a point of interest on a map and moving it on-screen. Future versions should work with different tasks and should be relevant for any situation. Enabling mid-air pointing for a smartwatch could open up to different possibilities in terms of user engagement.

6.2.3 Development of new mid-air interaction techniques

In this thesis I presented only the most common situation, panning through off-screen content. By developing different mechanisms for example zooming and rotating, it would be interesting to investigate the performance improvements or deteriorations in completion times, number of clutches or other factors. Implementing more gestures could also improve the usability of this prototype. Recognizing different gestures could help the users interact with granularity and could offer much tighter control in their tasks.

7 CONCLUSION

In this thesis, I have presented my findings from using off-screen mid-air pointing as an interaction technique for a smartwatch display. The goal was to investigate the performance and ease of use aspects of interacting with a smartwatch with much larger interaction space. A prototype has been developed for carrying out the research on the mid-air pointing technique.

Current research has focused extensively on mid-air interaction with larger screens such as wall sized displays and tablets. Some researchers investigated the use of mid-air pointing for smartphones and PDAs, but very few have tackled off-screen space for small sized displays, such as smartwatches. This inspired me to investigate off-screen mid-air pointing for smartwatches. Smartwatches have very limited touch areas and it would make sense that touch input wouldn't be the best interaction method for such a device. Thus, by creating the prototype presented in this thesis, I tried to understand and model how mid-air interaction would compare to the overly mainstream touch interface.

A user study has been done to compare mid-air pointing to the traditional touch interaction method and to explore the overall usability of my method. The experiment was composed of docking tasks with random initial starting points, categorized by distance (short, medium, far). During the experiment positional data, task completion time and number of interactions have been recorded. Participants had to complete a questionnaire, that is used to determine how the interaction methods compared to each other, in respects to fatigue, ease of use and overall opinions of the methods in general.

The results showed that the mid-air interaction method performed worse in terms of completion times and there was *no significant difference in fatigue ratings between mid-air and touch input. Mid-air pointing turned out to use less interact*ions, mainly due to the fact that the user's had a larger area of input. Overall, *the mid-air interaction did not perform better than touch* and my recommendation would be that future work is needed for improving the current prototype.

In conclusion, I have developed a prototype that uses motion tracking cameras to take advantage of off-screen space to improve the current interaction methods that are used in the industry for smartwatches. Although the results were not entirely favorable, I have suggested that improvements in tracking and minimizing delay would greatly impact the performance of the current implementation.

8 BIBLIOGRAPHY

[1] K. F. Haven, *100 greatest science inventions of all time*. Westport, Conn. [u.a.]: Libraries Unlimited, 2006.

[2] D. Boettcher. (2006). *The "invention" of the wristwatch*. Available: http://www.vintagewatchstraps.com/wristwatchinvention.php

[3] B. Hochet, *Integrated circuit design, power and timing modeling, optimization and simulation, 12th International Workshop, PATMOS 2002, Seville, Spain, September 11-13, 2002, proceedings*. Berlin: Springer, 2002.

[4] P. Doensen. (2010). *Q.5 Watches with Memory and Database*. Available: https://doensen.home.xs4all.nl/q5.html

[5] J. C. Ho, Brandon. (2015). *The Apple Watch Review: Apple S1 Analysis*. Available: http://www.anandtech.com/show/9381/the-apple-watch-review/3

[6] K. A. Siek, Y. Rogers, and K. H. Connelly, "Fat Finger Worries: How Older and Younger Users Physically Interact with PDAs," in *Human-Computer Interaction - INTERACT 2005: IFIP TC13 International Conference, Rome, Italy, September 12-16, 2005. Proceedings*, M. F. Costabile and F. Paternò, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 267-280.

[7] C. Forlines, D. Vogel, and R. Balakrishnan, "HybridPointing: fluid switching between absolute and relative pointing with a direct input device," presented at the Proceedings of the 19th annual ACM symposium on User interface software and technology, Montreux, Switzerland, 2006.

[8] H. Xia, T. Grossman, and G. Fitzmaurice, "NanoStylus: Enhancing Input on Ultra-Small Displays with a Finger-Mounted Stylus," presented at the Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, Daegu, Kyungpook, Republic of Korea, 2015.

[9] J. O. Wobbrock, B. A. Myers, and H. H. Aung, "The performance of hand postures in front- and back-of-device interaction for mobile computing," *International Journal of Human-Computer Studies,* vol. 66, pp. 857-875, 2008.

[10] D. Avrahami, J. O. Wobbrock, and S. Izadi, "Portico: tangible interaction on and around a tablet," presented at the Proceedings of the 24th annual ACM symposium on User interface software and technology, Santa Barbara, California, USA, 2011.

[11] T. Miyaki and J. Rekimoto, "GraspZoom: zooming and scrolling control model for single-handed mobile interaction," presented at the Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services, Bonn, Germany, 2009.

[12] W. V. Vlaenderen, J. Brulmans, J. Vermeulen, J. Sch, #246, and ning, "WatchMe: A Novel Input Method Combining a Smartwatch and Bimanual Interaction," presented at the Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems, Seoul, Republic of Korea, 2015.

[13] H. Ketabdar, K. A. Y, #252, ksel, and M. Roshandel, "MagiTact: interaction with mobile devices based on compass (magnetic) sensor," presented at the Proceedings of the 15th international conference on Intelligent user interfaces, Hong Kong, China, 2010.

[14] I. Oakley, D. Lee, M. R. Islam, and A. Esteves, "Beats: Tapping Gestures for Smart Watches," presented at the Proceedings of the 33rd Annual ACM

Conference on Human Factors in Computing Systems, Seoul, Republic of Korea, 2015.
[15] H. Wen, J. R. Rojas, and A. K. Dey, "Serendipity: Finger Gesture Recognition using an Off-the-Shelf Smartwatch," presented at the Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, Santa Clara, California, USA, 2016.
[16] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota, "FingerIO: Using Active Sonar for Fine-Grained Finger Tracking," presented at the Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, Santa Clara, California, USA, 2016.
[17] C. Harrison and S. E. Hudson, "Abracadabra: wireless, high-precision, and unpowered finger input for very small mobile devices," presented at the Proceedings of the 22nd annual ACM symposium on User interface software and technology, Victoria, BC, Canada, 2009.
[18] X.-D. Yang, K. Hasan, N. Bruce, and P. Irani, "Surround-see: enabling peripheral vision on smartphones during active use," presented at the Proceedings of the 26th annual ACM symposium on User interface software and technology, St. Andrews, Scotland, United Kingdom, 2013.
[19] J. Knibbe, D. M. Plasencia, C. Bainbridge, C.-K. Chan, J. Wu, T. Cable*, et al.*, "Extending interaction for smart watches: enabling bimanual around device control," presented at the Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems, Toronto, Ontario, Canada, 2014.
[20] S. Kratz and M. Rohs, "Hoverflow: exploring around-device interaction with IR distance sensors," presented at the Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services, Bonn, Germany, 2009.
[21] A. Butler, S. Izadi, and S. Hodges, "SideSight: multi-"touch" interaction around small devices," presented at the Proceedings of the 21st annual ACM symposium on User interface software and technology, Monterey, CA, USA, 2008.
[22] B. Kaufmann and D. Ahlström, "Studying spatial memory and map navigation performance on projector phones with peephole interaction," ed, 2013, pp. 3173-3176.
[23] G. Fitzmaurice, "Situated information spaces and spatially aware palmtop computers," *Communications of the ACM,* vol. 36, pp. 39-49, 1993.
[24] F. Kerber, A. Kr, #252, ger, M. L, #246*, et al.*, "Investigating the effectiveness of peephole interaction for smartwatches in a map navigation task," presented at the Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services, Toronto, ON, Canada, 2014.
[25] M. Pahud, K. Hinckley, S. Iqbal, A. Sellen, and B. Buxton, "Toward compound navigation tasks on mobiles via spatial manipulation," ed, 2013, pp. 113-122.
[26] K. Hasan, D. Ahlstr, #246, and P. P. Irani, "Comparing Direct Off-Screen Pointing, Peephole, and Flick & Pinch Interaction for Map Navigation," presented at the Proceedings of the 3rd ACM Symposium on Spatial User Interaction, Los Angeles, California, USA, 2015.
[27] P. Campos, N. Graham, J. Jorge, N. Nunes, P. Palanque, and M. Winckler, *Human-Computer Interaction -- INTERACT 2011: 13th IFIP TC 13 International Conference, Lisbon, Portugal, September 5-9, 2011, Proceedings*: Springer Berlin Heidelberg, 2011.

[28] Y. Guiard, M. Beaudouin-Lafon, J. Bastin, D. Pasveer, and S. Zhai, "View size and pointing difficulty in multi-scale navigation," presented at the Proceedings of the working conference on Advanced visual interfaces, Gallipoli, Italy, 2004.
[29] P. Baudisch and R. Rosenholtz, "Halo: a technique for visualizing off-screen objects," presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Ft. Lauderdale, Florida, USA, 2003.
[30] D. Vogel and R. Balakrishnan, "Distant freehand pointing and clicking on very large, high resolution displays," presented at the Proceedings of the 18th annual ACM symposium on User interface software and technology, Seattle, WA, USA, 2005.
[31] A. Cockburn, A. Karlson, and B. B. Bederson, "A review of overview+detail, zooming, and focus+context interfaces," *ACM Comput. Surv.,* vol. 41, pp. 1-31, 2009.
[32] R. Stoakley, M. Conway, and R. Pausch, "Virtual reality on a WIM: interactive worlds in miniature," ed, 1995, pp. 265-272.
[33] S. Gustafson, P. Baudisch, C. Gutwin, and P. Irani, "Wedge: clutter-free visualization of off-screen locations," presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, 2008.
[34] P. T. Zellweger, J. D. Mackinlay, L. Good, M. Stefik, and P. Baudisch, "City lights: contextual views in minimal space," presented at the CHI '03 Extended Abstracts on Human Factors in Computing Systems, Ft. Lauderdale, Florida, USA, 2003.
[35] P. Irani, C. Gutwin, G. Partridge, and M. Nezhadasl, "Techniques for Interacting with Off-Screen Content," in *Human-Computer Interaction – INTERACT 2007: 11th IFIP TC 13 International Conference, Rio de Janeiro, Brazil, September 10-14, 2007, Proceedings, Part II*, C. Baranauskas, P. Palanque, J. Abascal, and S. D. J. Barbosa, Eds., ed Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 234-249.
[36] K. Jungsoo Kim, T. Jiasheng He, T. Lyons, and T. Starner, "The Gesture Watch: A Wireless Contact-free Gesture based Wrist Interface," ed, 2007, pp. 15-22.
[37] B. Ens, D. Ahlstr, #246, A. Cockburn, and P. Irani, "Characterizing user performance with assisted direct off-screen pointing," presented at the Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services, Stockholm, Sweden, 2011.
[38] K. Hasan, D. Ahlstr, #246, and P. Irani, "Ad-binning: leveraging around device space for storing, browsing and retrieving mobile device content," presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Paris, France, 2013.
[39] A. Cockburn, P. Quinn, C. Gutwin, G. Ramos, and J. Looser, "Air pointing: Design and evaluation of spatial target acquisition with and without visual feedback," *International Journal of Human-Computer Studies,* vol. 69, pp. 401-414, 2011.
[40] B. Jones, R. Sodhi, D. Forsyth, B. Bailey, and G. Maciocci, "Around device interaction for multiscale navigation," presented at the Proceedings of the 14th international conference on Human-computer interaction with mobile devices and services, San Francisco, California, USA, 2012.
[41] K. Matsumura, "Studying User-Defined Gestures Toward Off the Screen Interactions," presented at the Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces, Madeira, Portugal, 2015.

[42] A. Markussen, S. Boring, M. R. Jakobsen, K. Hornb, and #230, "Off-Limits: Interacting Beyond the Boundaries of Large Displays," presented at the Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, Santa Clara, California, USA, 2016.
[43] . *OptiTrack Flex 3 specification page*. Available: http://optitrack.com/products/flex-3/specs.html
[44] D. Schmidt. (2014). *Short Review Simvalley Mobile AW-414.Go SmartWatch*. Available: http://www.notebookcheck.net/Short-Review-Simvalley-Mobile-AW-414-Go-SmartWatch.108427.0.html
[45] G. Casiez, N. Roussel, and D. Vogel, "1 € filter: a simple speed-based low-pass filter for noisy input in interactive systems," presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Austin, Texas, USA, 2012.
[46] I. O. f. Standardization, "ISO 9241-9:2000 - Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 9: Requirements for non-keyboard input devices," 2000.
[47] N. J. Gotelli and A. M. Ellison, *A Primer of Ecological Statistics*: Macmillan Education, 2013.
[48] R. R. Sokal and F. J. Rohlf, "Biometry: the principles of statistics in biological research," ed: New York, NY: WH Freeman and Co, 1995.

APPENDIX A

The appendix B contains the questionnaire that has been handed out to the participants of the user experiment.

Mid-air Pointing Interaction

Demographic Information & Device Assessment

Participant ID

____

DEMOGRAPHIC INFORMATION

Age:		**Gender:**		**Optical Aid:**		**Handedness:**	

Do you own a mobile touch screen device (e.g., iPhone/iPod)? Yes ☐ No ☐

Do you own a mobile smartwatch (e.g., Apple Watch/Moto 360)? Yes ☐ No ☐

Have you used a mobile touch-based device before? Yes ☐ No ☐

Have you used a map application before? Yes ☐ No ☐

Have you experienced other mid-air pointing technologies before? Yes ☐ No ☐

How do you rate your experience with touch-based mobile devices?

Highly inexperienced ☐ ☐ ☐ ☐ ☐ Highly experienced

DEVICE ASSESSMENT (1)

Touch+Inertia (i.e, pan & scroll directly using touch with inertia enabled)

1. **Smoothness during operation was:**
   Very rough ☐ ☐ ☐ ☐ ☐ Very smooth

2. **Operation speed was:**
   Too fast ☐ ☐ ☐ ☐ ☐ Too slow

3. **Finger fatigue:**
   None ☐ ☐ ☐ ☐ ☐ Very high

4. **Wrist fatigue:**
   None ☐ ☐ ☐ ☐ ☐ Very high

5. **General comfort:**
   Very uncomfortable ☐ ☐ ☐ ☐ ☐ Very comfortable

6. **Overall, the input method was:**
   Very difficult to use ☐ ☐ ☐ ☐ ☐ Very easy to use

Please state your own comments (advantages / disadvantages) below:

DEVICE ASSESSMENT (2)

"Mid-air" (i.e, panning through the interface using mid-air pointing)

1. **Smoothness during operation was:**
   Very rough ☐ ☐ ☐ ☐ ☐ Very smooth

2. **Operation speed was:**
   Too fast ☐ ☐ ☐ ☐ ☐ Too slow

3. **Finger fatigue:**
   None ☐ ☐ ☐ ☐ ☐ Very high

4. **Wrist fatigue:**
   None ☐ ☐ ☐ ☐ ☐ Very high

5. **General comfort:**
   Very uncomfortable ☐ ☐ ☐ ☐ ☐ Very comfortable

6. **Overall, the input method was:**
   Very difficult to use ☐ ☐ ☐ ☐ ☐ Very easy to use

Please state your own comments (advantages / disadvantages) below:

RANKING

Please rank the aforementioned systems

Touch+Inertia: _______

Mid-air pointing: _______

Additional general feedback (optional):

Appendix B

Appendix B contains important code snippets that have been used in the development of the prototype used in this thesis.

The following code snippet is the script that enables inertia scrolling when using the *touch+inertia* interaction method.

```
using UnityEngine;
public class Inertia : MonoBehaviour
{
    private Vector3 _screenPoint;
    private Vector3 _offset;
    private Vector3 _curScreenPoint;
    private Vector3 _curPosition;
    private Vector3 _velocity;
    private bool _underInertia;
    private float _time = 0.0f;
    public float SmoothTime;
    void Update()
    {
        if (_underInertia && _time <= SmoothTime)
        {
            transform.localPosition += new Vector3(_velocity.x, 0, _velocity.z);
            _velocity = Vector3.Lerp(_velocity, Vector3.zero, _time);
            _time += Time.smoothDeltaTime;
        }
        else
        {
            _underInertia = false;
            _time = 0.0f;
        }
    }
    void OnMouseDown()
    {
        _underInertia = false;
    }
    void OnMouseDrag()
    {
        Vector3 _prevPosition = _curPosition;
        _curScreenPoint = new Vector3(Input.mousePosition.x, Input.mousePosition.y,
_screenPoint.z);
        _curPosition = Camera.main.ScreenToWorldPoint(_curScreenPoint) + _offset;
        _velocity = _curPosition - _prevPosition;
    }
    void OnMouseUp()
    {
        _underInertia = true;
    }
}
```

APPENDIX C.1

contrast	estimate	SE	df	t.ratio	p.value
midair,long,E - midair,long,N	0.068016555	0.05874203	364	1.158	0.2477
midair,long,E - midair,long,NE	-0.173258472	0.05874203	364	-2.949	0.0034
midair,long,E - midair,long,NW	-0.014479930	0.05874203	364	-0.247	0.8054
midair,long,E - midair,long,S	0.112505906	0.05874203	364	1.915	0.0562
midair,long,E - midair,long,SE	-0.111205630	0.05874203	364	-1.893	0.0591
midair,long,E - midair,long,SW	-0.093040287	0.05874203	364	-1.584	0.1141
midair,long,E - midair,long,W	0.121790062	0.05874203	364	2.073	0.0388
touch+inertia,long,E - touch+inertia,long,N	0.068016555	0.05874203	364	1.158	0.2477
touch+inertia,long,E - touch+inertia,long,NE	-0.173258472	0.05874203	364	-2.949	0.0034
touch+inertia,long,E - touch+inertia,long,NW	-0.014479930	0.05874203	364	-0.247	0.8054
touch+inertia,long,E - touch+inertia,long,S	0.112505906	0.05874203	364	1.915	0.0562
touch+inertia,long,E - touch+inertia,long,SE	-0.111205630	0.05874203	364	-1.893	0.0591
touch+inertia,long,E - touch+inertia,long,SW	-0.093040287	0.05874203	364	-1.584	0.1141
touch+inertia,long,E - touch+inertia,long,W	0.121790062	0.05874203	364	2.073	0.0388
midair,medium,E - midair,medium,N	0.068016555	0.05874203	364	1.158	0.2477
midair,medium,E - midair,medium,NE	-0.173258472	0.05874203	364	-2.949	0.0034
midair,medium,E - midair,medium,NW	-0.014479930	0.05874203	364	-0.247	0.8054
midair,medium,E - midair,medium,S	0.112505906	0.05874203	364	1.915	0.0562
midair,medium,E - midair,medium,SE	-0.111205630	0.05874203	364	-1.893	0.0591
midair,medium,E - midair,medium,SW	-0.093040287	0.05874203	364	-1.584	0.1141
midair,medium,E - midair,medium,W	0.121790062	0.05874203	364	2.073	0.0388
touch+inertia,medium,E - touch+inertia,medium,N	0.068016555	0.05874203	364	1.158	0.2477
touch+inertia,medium,E - touch+inertia,medium,NE	-0.173258472	0.05874203	364	-2.949	0.0034
touch+inertia,medium,E - touch+inertia,medium,NW	-0.014479930	0.05874203	364	-0.247	0.8054
touch+inertia,medium,E - touch+inertia,medium,S	0.112505906	0.05874203	364	1.915	0.0562
touch+inertia,medium,E - touch+inertia,medium,SE	-0.111205630	0.05874203	364	-1.893	0.0591
touch+inertia,medium,E - touch+inertia,medium,SW	-0.093040287	0.05874203	364	-1.584	0.1141
touch+inertia,medium,E - touch+inertia,medium,W	0.121790062	0.05874203	364	2.073	0.0388
midair,short,E - midair,short,N	0.068016555	0.05874203	364	1.158	0.2477
midair,short,E - midair,short,NE	-0.173258472	0.05874203	364	-2.949	0.0034
midair,short,E - midair,short,NW	-0.014479930	0.05874203	364	-0.247	0.8054
midair,short,E - midair,short,S	0.112505906	0.05874203	364	1.915	0.0562
midair,short,E - midair,short,SE	-0.111205630	0.05874203	364	-1.893	0.0591
midair,short,E - midair,short,SW	-0.093040287	0.05874203	364	-1.584	0.1141
midair,short,E - midair,short,W	0.121790062	0.05874203	364	2.073	0.0388
touch+inertia,short,E - touch+inertia,short,N	0.068016555	0.05874203	364	1.158	0.2477
touch+inertia,short,E - touch+inertia,short,NE	-0.173258472	0.05874203	364	-2.949	0.0034
touch+inertia,short,E - touch+inertia,short,NW	-0.014479930	0.05874203	364	-0.247	0.8054
touch+inertia,short,E - touch+inertia,short,S	0.112505906	0.05874203	364	1.915	0.0562
touch+inertia,short,E - touch+inertia,short,SE	-0.111205630	0.05874203	364	-1.893	0.0591
touch+inertia,short,E - touch+inertia,short,SW	-0.093040287	0.05874203	364	-1.584	0.1141
touch+inertia,short,E - touch+inertia,short,W	0.121790062	0.05874203	364	2.073	0.0388
touch+inertia,short,N - touch+inertia,short,NE	-0.241275027	0.05874203	364	-4.107	<.0001
touch+inertia,short,N - touch+inertia,short,NW	-0.082496485	0.05874203	364	-1.404	0.1611
touch+inertia,short,N - touch+inertia,short,S	0.044489351	0.05874203	364	0.757	0.4493
touch+inertia,short,N - touch+inertia,short,SE	-0.179222185	0.05874203	364	-3.051	0.0024
touch+inertia,short,N - touch+inertia,short,SW	-0.161056842	0.05874203	364	-2.742	0.0064
touch+inertia,short,N - touch+inertia,short,W	0.053773507	0.05874203	364	0.915	0.3606
midair,long,N - midair,long,NE	-0.241275027	0.05874203	364	-4.107	<.0001
midair,long,N - midair,long,NW	-0.082496485	0.05874203	364	-1.404	0.1611
midair,long,N - midair,long,S	0.044489351	0.05874203	364	0.757	0.4493
midair,long,N - midair,long,SE	-0.179222185	0.05874203	364	-3.051	0.0024
midair,long,N - midair,long,SW	-0.161056842	0.05874203	364	-2.742	0.0064
midair,long,N - midair,long,W	0.053773507	0.05874203	364	0.915	0.3606
touch+inertia,long,N - touch+inertia,long,NE	-0.241275027	0.05874203	364	-4.107	<.0001
touch+inertia,long,N - touch+inertia,long,NW	-0.082496485	0.05874203	364	-1.404	0.1611
touch+inertia,long,N - touch+inertia,long,S	0.044489351	0.05874203	364	0.757	0.4493
touch+inertia,long,N - touch+inertia,long,SE	-0.179222185	0.05874203	364	-3.051	0.0024
touch+inertia,long,N - touch+inertia,long,SW	-0.161056842	0.05874203	364	-2.742	0.0064
touch+inertia,long,N - touch+inertia,long,W	0.053773507	0.05874203	364	0.915	0.3606
midair,medium,N - midair,medium,NE	-0.241275027	0.05874203	364	-4.107	<.0001
midair,medium,N - midair,medium,NW	-0.082496485	0.05874203	364	-1.404	0.1611
midair,medium,N - midair,medium,S	0.044489351	0.05874203	364	0.757	0.4493
midair,medium,N - midair,medium,SE	-0.179222185	0.05874203	364	-3.051	0.0024
midair,medium,N - midair,medium,SW	-0.161056842	0.05874203	364	-2.742	0.0064
midair,medium,N - midair,medium,W	0.053773507	0.05874203	364	0.915	0.3606
touch+inertia,medium,N - touch+inertia,medium,NE	-0.241275027	0.05874203	364	-4.107	<.0001
touch+inertia,medium,N - touch+inertia,medium,NW	-0.082496485	0.05874203	364	-1.404	0.1611
touch+inertia,medium,N - touch+inertia,medium,S	0.044489351	0.05874203	364	0.757	0.4493
touch+inertia,medium,N - touch+inertia,medium,SE	-0.179222185	0.05874203	364	-3.051	0.0024
touch+inertia,medium,N - touch+inertia,medium,SW	-0.161056842	0.05874203	364	-2.742	0.0064
touch+inertia,medium,N - touch+inertia,medium,W	0.053773507	0.05874203	364	0.915	0.3606
midair,short,N - midair,short,NE	-0.241275027	0.05874203	364	-4.107	<.0001
midair,short,N - midair,short,NW	-0.082496485	0.05874203	364	-1.404	0.1611
midair,short,N - midair,short,S	0.044489351	0.05874203	364	0.757	0.4493

midair,short,N - midair,short,SE	-0.179222185	0.05874203	364	-3.051	0.0024
midair,short,N - midair,short,SW	-0.161056842	0.05874203	364	-2.742	0.0064
midair,short,N - midair,short,W	0.053773507	0.05874203	364	0.915	0.3606
midair,long,NE - midair,long,NW	0.158778542	0.05874203	364	2.703	0.0072
midair,long,NE - midair,long,S	0.285764378	0.05874203	364	4.865	<.0001
midair,long,NE - midair,long,SE	0.062052842	0.05874203	364	1.056	0.2915
midair,long,NE - midair,long,SW	0.080218184	0.05874203	364	1.366	0.1729
midair,long,NE - midair,long,W	0.295048534	0.05874203	364	5.023	<.0001
touch+inertia,long,NE - touch+inertia,long,NW	0.158778542	0.05874203	364	2.703	0.0072
touch+inertia,long,NE - touch+inertia,long,S	0.285764378	0.05874203	364	4.865	<.0001
touch+inertia,long,NE - touch+inertia,long,SE	0.062052842	0.05874203	364	1.056	0.2915
touch+inertia,long,NE - touch+inertia,long,SW	0.080218184	0.05874203	364	1.366	0.1729
touch+inertia,long,NE - touch+inertia,long,W	0.295048534	0.05874203	364	5.023	<.0001
midair,medium,NE - midair,medium,NW	0.158778542	0.05874203	364	2.703	0.0072
midair,medium,NE - midair,medium,S	0.285764378	0.05874203	364	4.865	<.0001
midair,medium,NE - midair,medium,SE	0.062052842	0.05874203	364	1.056	0.2915
midair,medium,NE - midair,medium,SW	0.080218184	0.05874203	364	1.366	0.1729
midair,medium,NE - midair,medium,W	0.295048534	0.05874203	364	5.023	<.0001
touch+inertia,medium,NE - touch+inertia,medium,NW	0.158778542	0.05874203	364	2.703	0.0072
touch+inertia,medium,NE - touch+inertia,medium,S	0.285764378	0.05874203	364	4.865	<.0001
touch+inertia,medium,NE - touch+inertia,medium,SE	0.062052842	0.05874203	364	1.056	0.2915
touch+inertia,medium,NE - touch+inertia,medium,SW	0.080218184	0.05874203	364	1.366	0.1729
touch+inertia,medium,NE - touch+inertia,medium,W	0.295048534	0.05874203	364	5.023	<.0001
midair,short,NE - midair,short,NW	0.158778542	0.05874203	364	2.703	0.0072
midair,short,NE - midair,short,S	0.285764378	0.05874203	364	4.865	<.0001
midair,short,NE - midair,short,SE	0.062052842	0.05874203	364	1.056	0.2915
midair,short,NE - midair,short,SW	0.080218184	0.05874203	364	1.366	0.1729
midair,short,NE - midair,short,W	0.295048534	0.05874203	364	5.023	<.0001
touch+inertia,short,NE - touch+inertia,short,NW	0.158778542	0.05874203	364	2.703	0.0072
touch+inertia,short,NE - touch+inertia,short,S	0.285764378	0.05874203	364	4.865	<.0001
touch+inertia,short,NE - touch+inertia,short,SE	0.062052842	0.05874203	364	1.056	0.2915
touch+inertia,short,NE - touch+inertia,short,SW	0.080218184	0.05874203	364	1.366	0.1729
touch+inertia,short,NE - touch+inertia,short,W	0.295048534	0.05874203	364	5.023	<.0001
midair,long,NW - midair,long,S	0.126985836	0.05874203	364	2.162	0.0313
midair,long,NW - midair,long,SE	-0.096725700	0.05874203	364	-1.647	0.1005
midair,long,NW - midair,long,SW	-0.078560357	0.05874203	364	-1.337	0.1819
midair,long,NW - midair,long,W	0.136269992	0.05874203	364	2.320	0.0209
touch+inertia,long,NW - touch+inertia,long,S	0.126985836	0.05874203	364	2.162	0.0313
touch+inertia,long,NW - touch+inertia,long,SE	-0.096725700	0.05874203	364	-1.647	0.1005
touch+inertia,long,NW - touch+inertia,long,SW	-0.078560357	0.05874203	364	-1.337	0.1819
touch+inertia,long,NW - touch+inertia,long,W	0.136269992	0.05874203	364	2.320	0.0209
midair,medium,NW - midair,medium,S	0.126985836	0.05874203	364	2.162	0.0313
midair,medium,NW - midair,medium,SE	-0.096725700	0.05874203	364	-1.647	0.1005
midair,medium,NW - midair,medium,SW	-0.078560357	0.05874203	364	-1.337	0.1819
midair,medium,NW - midair,medium,W	0.136269992	0.05874203	364	2.320	0.0209
touch+inertia,medium,NW - touch+inertia,medium,S	0.126985836	0.05874203	364	2.162	0.0313
touch+inertia,medium,NW - touch+inertia,medium,SE	-0.096725700	0.05874203	364	-1.647	0.1005
touch+inertia,medium,NW - touch+inertia,medium,SW	-0.078560357	0.05874203	364	-1.337	0.1819
touch+inertia,medium,NW - touch+inertia,medium,W	0.136269992	0.05874203	364	2.320	0.0209
midair,short,NW - midair,short,S	0.126985836	0.05874203	364	2.162	0.0313
midair,short,NW - midair,short,SE	-0.096725700	0.05874203	364	-1.647	0.1005
midair,short,NW - midair,short,SW	-0.078560357	0.05874203	364	-1.337	0.1819
midair,short,NW - midair,short,W	0.136269992	0.05874203	364	2.320	0.0209
touch+inertia,short,NW - touch+inertia,short,S	0.126985836	0.05874203	364	2.162	0.0313
touch+inertia,short,NW - touch+inertia,short,SE	-0.096725700	0.05874203	364	-1.647	0.1005
touch+inertia,short,NW - touch+inertia,short,SW	-0.078560357	0.05874203	364	-1.337	0.1819
touch+inertia,short,NW - touch+inertia,short,W	0.136269992	0.05874203	364	2.320	0.0209
midair,long,S - midair,long,SE	-0.223711536	0.05874203	364	-3.808	0.0002
midair,long,S - midair,long,SW	-0.205546193	0.05874203	364	-3.499	0.0005
midair,long,S - midair,long,W	0.009284156	0.05874203	364	0.158	0.8745
touch+inertia,long,S - touch+inertia,long,SE	-0.223711536	0.05874203	364	-3.808	0.0002
touch+inertia,long,S - touch+inertia,long,SW	-0.205546193	0.05874203	364	-3.499	0.0005
touch+inertia,long,S - touch+inertia,long,W	0.009284156	0.05874203	364	0.158	0.8745
midair,medium,S - midair,medium,SE	-0.223711536	0.05874203	364	-3.808	0.0002
midair,medium,S - midair,medium,SW	-0.205546193	0.05874203	364	-3.499	0.0005
midair,medium,S - midair,medium,W	0.009284156	0.05874203	364	0.158	0.8745
touch+inertia,medium,S - touch+inertia,medium,SE	-0.223711536	0.05874203	364	-3.808	0.0002
touch+inertia,medium,S - touch+inertia,medium,SW	-0.205546193	0.05874203	364	-3.499	0.0005
touch+inertia,medium,S - touch+inertia,medium,W	0.009284156	0.05874203	364	0.158	0.8745
midair,short,S - midair,short,SE	-0.223711536	0.05874203	364	-3.808	0.0002
midair,short,S - midair,short,SW	-0.205546193	0.05874203	364	-3.499	0.0005
midair,short,S - midair,short,W	0.009284156	0.05874203	364	0.158	0.8745
touch+inertia,short,S - touch+inertia,short,SE	-0.223711536	0.05874203	364	-3.808	0.0002
touch+inertia,short,S - touch+inertia,short,SW	-0.205546193	0.05874203	364	-3.499	0.0005
touch+inertia,short,S - touch+inertia,short,W	0.009284156	0.05874203	364	0.158	0.8745
midair,long,SE - midair,long,SW	0.018165342	0.05874203	364	0.309	0.7573
midair,long,SE - midair,long,W	0.232995692	0.05874203	364	3.966	0.0001
touch+inertia,long,SE - touch+inertia,long,SW	0.018165342	0.05874203	364	0.309	0.7573
touch+inertia,long,SE - touch+inertia,long,W	0.232995692	0.05874203	364	3.966	0.0001

midair,medium,SE - midair,medium,SW	0.018165342	0.05874203	364	0.309	0.7573
midair,medium,SE - midair,medium,W	0.232995692	0.05874203	364	3.966	0.0001
touch+inertia,medium,SE - touch+inertia,medium,SW	0.018165342	0.05874203	364	0.309	0.7573
touch+inertia,medium,SE - touch+inertia,medium,W	0.232995692	0.05874203	364	3.966	0.0001
midair,short,SE - midair,short,SW	0.018165342	0.05874203	364	0.309	0.7573
midair,short,SE - midair,short,W	0.232995692	0.05874203	364	3.966	0.0001
touch+inertia,short,SE - touch+inertia,short,SW	0.018165342	0.05874203	364	0.309	0.7573
touch+inertia,short,SE - touch+inertia,short,W	0.232995692	0.05874203	364	3.966	0.0001
midair,long,SW - midair,long,W	0.214830350	0.05874203	364	3.657	0.0003
touch+inertia,long,SW - touch+inertia,long,W	0.214830350	0.05874203	364	3.657	0.0003
midair,medium,SW - midair,medium,W	0.214830350	0.05874203	364	3.657	0.0003
touch+inertia,medium,SW - touch+inertia,medium,W	0.214830350	0.05874203	364	3.657	0.0003
midair,short,SW - midair,short,W	0.214830350	0.05874203	364	3.657	0.0003
touch+inertia,short,SW - touch+inertia,short,W	0.214830350	0.05874203	364	3.657	0.0003

Appendix C.2

contrast	estimate	SE	df	t.ratio	p.value
midair,long,E - midair,long,N	-0.0066955740	0.006064835	366.01	-1.104	0.2703
midair,long,E - midair,long,NE	0.0108814417	0.006064835	366.01	1.794	0.0736
midair,long,E - midair,long,NW	0.0080039914	0.006064835	366.01	1.320	0.1877
midair,long,E - midair,long,S	-0.0103619155	0.006064835	366.01	-1.709	0.0884
midair,long,E - midair,long,SE	0.0121597303	0.006064835	366.01	2.005	0.0457
midair,long,E - midair,long,SW	0.0055777071	0.006064835	366.01	0.920	0.3583
midair,long,E - midair,long,W	-0.0159804617	0.006064835	366.01	-2.635	0.0088
touch+inertia,long,E - touch+inertia,long,N	-0.0066955740	0.006064835	366.01	-1.104	0.2703
touch+inertia,long,E - touch+inertia,long,NE	0.0108814417	0.006064835	366.01	1.794	0.0736
touch+inertia,long,E - touch+inertia,long,NW	0.0080039914	0.006064835	366.01	1.320	0.1877
touch+inertia,long,E - touch+inertia,long,S	-0.0103619155	0.006064835	366.01	-1.709	0.0884
touch+inertia,long,E - touch+inertia,long,SE	0.0121597303	0.006064835	366.01	2.005	0.0457
touch+inertia,long,E - touch+inertia,long,SW	0.0055777071	0.006064835	366.01	0.920	0.3583
touch+inertia,long,E - touch+inertia,long,W	-0.0159804617	0.006064835	366.01	-2.635	0.0088
midair,medium,E - midair,medium,N	-0.0066955740	0.006064835	366.01	-1.104	0.2703
midair,medium,E - midair,medium,NE	0.0108814417	0.006064835	366.01	1.794	0.0736
midair,medium,E - midair,medium,NW	0.0080039914	0.006064835	366.01	1.320	0.1877
midair,medium,E - midair,medium,S	-0.0103619155	0.006064835	366.01	-1.709	0.0884
midair,medium,E - midair,medium,SE	0.0121597303	0.006064835	366.01	2.005	0.0457
midair,medium,E - midair,medium,SW	0.0055777071	0.006064835	366.01	0.920	0.3583
midair,medium,E - midair,medium,W	-0.0159804617	0.006064835	366.01	-2.635	0.0088
touch+inertia,medium,E - touch+inertia,medium,N	-0.0066955740	0.006064835	366.01	-1.104	0.2703
touch+inertia,medium,E - touch+inertia,medium,NE	0.0108814417	0.006064835	366.01	1.794	0.0736
touch+inertia,medium,E - touch+inertia,medium,NW	0.0080039914	0.006064835	366.01	1.320	0.1877
touch+inertia,medium,E - touch+inertia,medium,S	-0.0103619155	0.006064835	366.01	-1.709	0.0884
touch+inertia,medium,E - touch+inertia,medium,SE	0.0121597303	0.006064835	366.01	2.005	0.0457
touch+inertia,medium,E - touch+inertia,medium,SW	0.0055777071	0.006064835	366.01	0.920	0.3583
touch+inertia,medium,E - touch+inertia,medium,W	-0.0159804617	0.006064835	366.01	-2.635	0.0088
midair,short,E - midair,short,N	-0.0066955740	0.006064835	366.01	-1.104	0.2703
midair,short,E - midair,short,NE	0.0108814417	0.006064835	366.01	1.794	0.0736
midair,short,E - midair,short,NW	0.0080039914	0.006064835	366.01	1.320	0.1877
midair,short,E - midair,short,S	-0.0103619155	0.006064835	366.01	-1.709	0.0884
midair,short,E - midair,short,SE	0.0121597303	0.006064835	366.01	2.005	0.0457
midair,short,E - midair,short,SW	0.0055777071	0.006064835	366.01	0.920	0.3583
midair,short,E - midair,short,W	-0.0159804617	0.006064835	366.01	-2.635	0.0088
touch+inertia,short,E - touch+inertia,short,N	-0.0066955740	0.006064835	366.01	-1.104	0.2703
touch+inertia,short,E - touch+inertia,short,NE	0.0108814417	0.006064835	366.01	1.794	0.0736
touch+inertia,short,E - touch+inertia,short,NW	0.0080039914	0.006064835	366.01	1.320	0.1877
touch+inertia,short,E - touch+inertia,short,S	-0.0103619155	0.006064835	366.01	-1.709	0.0884
touch+inertia,short,E - touch+inertia,short,SE	0.0121597303	0.006064835	366.01	2.005	0.0457
touch+inertia,short,E - touch+inertia,short,SW	0.0055777071	0.006064835	366.01	0.920	0.3583
touch+inertia,short,E - touch+inertia,short,W	-0.0159804617	0.006064835	366.01	-2.635	0.0088
midair,long,N - midair,long,NE	0.0175770157	0.006064835	366.01	2.898	0.0040
midair,long,N - midair,long,NW	0.0146995654	0.006064835	366.01	2.424	0.0158
midair,long,N - midair,long,S	-0.0036663416	0.006064835	366.01	-0.605	0.5459
midair,long,N - midair,long,SE	0.0188553043	0.006064835	366.01	3.109	0.0020
midair,long,N - midair,long,SW	0.0122732810	0.006064835	366.01	2.024	0.0437
midair,long,N - midair,long,W	-0.0092848878	0.006064835	366.01	-1.531	0.1266
touch+inertia,long,N - touch+inertia,long,NE	0.0175770157	0.006064835	366.01	2.898	0.0040
touch+inertia,long,N - touch+inertia,long,NW	0.0146995654	0.006064835	366.01	2.424	0.0158
touch+inertia,long,N - touch+inertia,long,S	-0.0036663416	0.006064835	366.01	-0.605	0.5459
touch+inertia,long,N - touch+inertia,long,SE	0.0188553043	0.006064835	366.01	3.109	0.0020
touch+inertia,long,N - touch+inertia,long,SW	0.0122732810	0.006064835	366.01	2.024	0.0437
touch+inertia,long,N - touch+inertia,long,W	-0.0092848878	0.006064835	366.01	-1.531	0.1266
midair,medium,N - midair,medium,NE	0.0175770157	0.006064835	366.01	2.898	0.0040
midair,medium,N - midair,medium,NW	0.0146995654	0.006064835	366.01	2.424	0.0158
midair,medium,N - midair,medium,S	-0.0036663416	0.006064835	366.01	-0.605	0.5459
midair,medium,N - midair,medium,SE	0.0188553043	0.006064835	366.01	3.109	0.0020
midair,medium,N - midair,medium,SW	0.0122732810	0.006064835	366.01	2.024	0.0437
midair,medium,N - midair,medium,W	-0.0092848878	0.006064835	366.01	-1.531	0.1266
touch+inertia,medium,N - touch+inertia,medium,NE	0.0175770157	0.006064835	366.01	2.898	0.0040
touch+inertia,medium,N - touch+inertia,medium,NW	0.0146995654	0.006064835	366.01	2.424	0.0158
touch+inertia,medium,N - touch+inertia,medium,S	-0.0036663416	0.006064835	366.01	-0.605	0.5459
touch+inertia,medium,N - touch+inertia,medium,SE	0.0188553043	0.006064835	366.01	3.109	0.0020
touch+inertia,medium,N - touch+inertia,medium,SW	0.0122732810	0.006064835	366.01	2.024	0.0437
touch+inertia,medium,N - touch+inertia,medium,W	-0.0092848878	0.006064835	366.01	-1.531	0.1266
midair,short,N - midair,short,NE	0.0175770157	0.006064835	366.01	2.898	0.0040
midair,short,N - midair,short,NW	0.0146995654	0.006064835	366.01	2.424	0.0158
midair,short,N - midair,short,S	-0.0036663416	0.006064835	366.01	-0.605	0.5459
midair,short,N - midair,short,SE	0.0188553043	0.006064835	366.01	3.109	0.0020
midair,short,N - midair,short,SW	0.0122732810	0.006064835	366.01	2.024	0.0437
midair,short,N - midair,short,W	-0.0092848878	0.006064835	366.01	-1.531	0.1266
touch+inertia,short,N - touch+inertia,short,NE	0.0175770157	0.006064835	366.01	2.898	0.0040
touch+inertia,short,N - touch+inertia,short,NW	0.0146995654	0.006064835	366.01	2.424	0.0158
touch+inertia,short,N - touch+inertia,short,S	-0.0036663416	0.006064835	366.01	-0.605	0.5459
touch+inertia,short,N - touch+inertia,short,SE	0.0188553043	0.006064835	366.01	3.109	0.0020
touch+inertia,short,N - touch+inertia,short,SW	0.0122732810	0.006064835	366.01	2.024	0.0437
touch+inertia,short,N - touch+inertia,short,W	-0.0092848878	0.006064835	366.01	-1.531	0.1266
midair,long,NE - midair,long,NW	-0.0028774503	0.006064835	366.01	-0.474	0.6355
midair,long,NE - midair,long,S	-0.0212433573	0.006064835	366.01	-3.503	0.0005

midair,long,NE - midair,long,SE	0.0012782886	0.006064835	366.01	0.211	0.8332
midair,long,NE - midair,long,SW	-0.0053037346	0.006064835	366.01	-0.875	0.3824
midair,long,NE - midair,long,W	-0.0268619034	0.006064835	366.01	-4.429	<.0001
touch+inertia,long,NE - touch+inertia,long,NW	-0.0028774503	0.006064835	366.01	-0.474	0.6355
touch+inertia,long,NE - touch+inertia,long,S	-0.0212433573	0.006064835	366.01	-3.503	0.0005
touch+inertia,long,NE - touch+inertia,long,SE	0.0012782886	0.006064835	366.01	0.211	0.8332
touch+inertia,long,NE - touch+inertia,long,SW	-0.0053037346	0.006064835	366.01	-0.875	0.3824
touch+inertia,long,NE - touch+inertia,long,W	-0.0268619034	0.006064835	366.01	-4.429	<.0001
midair,medium,NE - midair,medium,NW	-0.0028774503	0.006064835	366.01	-0.474	0.6355
midair,medium,NE - midair,medium,S	-0.0212433573	0.006064835	366.01	-3.503	0.0005
midair,medium,NE - midair,medium,SE	0.0012782886	0.006064835	366.01	0.211	0.8332
midair,medium,NE - midair,medium,SW	-0.0053037346	0.006064835	366.01	-0.875	0.3824
midair,medium,NE - midair,medium,W	-0.0268619034	0.006064835	366.01	-4.429	<.0001
touch+inertia,medium,NE - touch+inertia,medium,NW	-0.0028774503	0.006064835	366.01	-0.474	0.6355
touch+inertia,medium,NE - touch+inertia,medium,S	-0.0212433573	0.006064835	366.01	-3.503	0.0005
touch+inertia,medium,NE - touch+inertia,medium,SE	0.0012782886	0.006064835	366.01	0.211	0.8332
touch+inertia,medium,NE - touch+inertia,medium,SW	-0.0053037346	0.006064835	366.01	-0.875	0.3824
touch+inertia,medium,NE - touch+inertia,medium,W	-0.0268619034	0.006064835	366.01	-4.429	<.0001
midair,short,NE - midair,short,NW	-0.0028774503	0.006064835	366.01	-0.474	0.6355
midair,short,NE - midair,short,S	-0.0212433573	0.006064835	366.01	-3.503	0.0005
midair,short,NE - midair,short,SE	0.0012782886	0.006064835	366.01	0.211	0.8332
midair,short,NE - midair,short,SW	-0.0053037346	0.006064835	366.01	-0.875	0.3824
midair,short,NE - midair,short,W	-0.0268619034	0.006064835	366.01	-4.429	<.0001
touch+inertia,short,NE - touch+inertia,short,NW	-0.0028774503	0.006064835	366.01	-0.474	0.6355
touch+inertia,short,NE - touch+inertia,short,S	-0.0212433573	0.006064835	366.01	-3.503	0.0005
touch+inertia,short,NE - touch+inertia,short,SE	0.0012782886	0.006064835	366.01	0.211	0.8332
touch+inertia,short,NE - touch+inertia,short,SW	-0.0053037346	0.006064835	366.01	-0.875	0.3824
touch+inertia,short,NE - touch+inertia,short,W	-0.0268619034	0.006064835	366.01	-4.429	<.0001
midair,long,NW - midair,long,S	-0.0183659070	0.006064835	366.01	-3.028	0.0026
midair,long,NW - midair,long,SE	0.0041557389	0.006064835	366.01	0.685	0.4936
midair,long,NW - midair,long,SW	-0.0024262844	0.006064835	366.01	-0.400	0.6893
midair,long,NW - midair,long,W	-0.0239844532	0.006064835	366.01	-3.955	0.0001
touch+inertia,long,NW - touch+inertia,long,S	-0.0183659070	0.006064835	366.01	-3.028	0.0026
touch+inertia,long,NW - touch+inertia,long,SE	0.0041557389	0.006064835	366.01	0.685	0.4936
touch+inertia,long,NW - touch+inertia,long,SW	-0.0024262844	0.006064835	366.01	-0.400	0.6893
touch+inertia,long,NW - touch+inertia,long,W	-0.0239844532	0.006064835	366.01	-3.955	0.0001
midair,medium,NW - midair,medium,S	-0.0183659070	0.006064835	366.01	-3.028	0.0026
midair,medium,NW - midair,medium,SE	0.0041557389	0.006064835	366.01	0.685	0.4936
midair,medium,NW - midair,medium,SW	-0.0024262844	0.006064835	366.01	-0.400	0.6893
midair,medium,NW - midair,medium,W	-0.0239844532	0.006064835	366.01	-3.955	0.0001
touch+inertia,medium,NW - touch+inertia,medium,S	-0.0183659070	0.006064835	366.01	-3.028	0.0026
touch+inertia,medium,NW - touch+inertia,medium,SE	0.0041557389	0.006064835	366.01	0.685	0.4936
touch+inertia,medium,NW - touch+inertia,medium,SW	-0.0024262844	0.006064835	366.01	-0.400	0.6893
touch+inertia,medium,NW - touch+inertia,medium,W	-0.0239844532	0.006064835	366.01	-3.955	0.0001
midair,short,NW - midair,short,S	-0.0183659070	0.006064835	366.01	-3.028	0.0026
midair,short,NW - midair,short,SE	0.0041557389	0.006064835	366.01	0.685	0.4936
midair,short,NW - midair,short,SW	-0.0024262844	0.006064835	366.01	-0.400	0.6893
midair,short,NW - midair,short,W	-0.0239844532	0.006064835	366.01	-3.955	0.0001
touch+inertia,short,NW - touch+inertia,short,S	-0.0183659070	0.006064835	366.01	-3.028	0.0026
touch+inertia,short,NW - touch+inertia,short,SE	0.0041557389	0.006064835	366.01	0.685	0.4936
touch+inertia,short,NW - touch+inertia,short,SW	-0.0024262844	0.006064835	366.01	-0.400	0.6893
touch+inertia,short,NW - touch+inertia,short,W	-0.0239844532	0.006064835	366.01	-3.955	0.0001
midair,long,S - midair,long,SE	0.0225216459	0.006064835	366.01	3.713	0.0002
midair,long,S - midair,long,SW	0.0159396226	0.006064835	366.01	2.628	0.0089
midair,long,S - midair,long,W	-0.0056185462	0.006064835	366.01	-0.926	0.3548
touch+inertia,long,S - touch+inertia,long,SE	0.0225216459	0.006064835	366.01	3.713	0.0002
touch+inertia,long,S - touch+inertia,long,SW	0.0159396226	0.006064835	366.01	2.628	0.0089
touch+inertia,long,S - touch+inertia,long,W	-0.0056185462	0.006064835	366.01	-0.926	0.3548
midair,medium,S - midair,medium,SE	0.0225216459	0.006064835	366.01	3.713	0.0002
midair,medium,S - midair,medium,SW	0.0159396226	0.006064835	366.01	2.628	0.0089
midair,medium,S - midair,medium,W	-0.0056185462	0.006064835	366.01	-0.926	0.3548
touch+inertia,medium,S - touch+inertia,medium,SE	0.0225216459	0.006064835	366.01	3.713	0.0002
touch+inertia,medium,S - touch+inertia,medium,SW	0.0159396226	0.006064835	366.01	2.628	0.0089
touch+inertia,medium,S - touch+inertia,medium,W	-0.0056185462	0.006064835	366.01	-0.926	0.3548
midair,short,S - midair,short,SE	0.0225216459	0.006064835	366.01	3.713	0.0002
midair,short,S - midair,short,SW	0.0159396226	0.006064835	366.01	2.628	0.0089
midair,short,S - midair,short,W	-0.0056185462	0.006064835	366.01	-0.926	0.3548
touch+inertia,short,S - touch+inertia,short,SE	0.0225216459	0.006064835	366.01	3.713	0.0002
touch+inertia,short,S - touch+inertia,short,SW	0.0159396226	0.006064835	366.01	2.628	0.0089
touch+inertia,short,S - touch+inertia,short,W	-0.0056185462	0.006064835	366.01	-0.926	0.3548
midair,long,SE - midair,long,SW	-0.0065820233	0.006064835	366.01	-1.085	0.2785
midair,long,SE - midair,long,W	-0.0281401921	0.006064835	366.01	-4.640	<.0001
touch+inertia,long,SE - touch+inertia,long,SW	-0.0065820233	0.006064835	366.01	-1.085	0.2785
touch+inertia,long,SE - touch+inertia,long,W	-0.0281401921	0.006064835	366.01	-4.640	<.0001
midair,medium,SE - midair,medium,SW	-0.0065820233	0.006064835	366.01	-1.085	0.2785
midair,medium,SE - midair,medium,W	-0.0281401921	0.006064835	366.01	-4.640	<.0001
touch+inertia,medium,SE - touch+inertia,medium,SW	-0.0065820233	0.006064835	366.01	-1.085	0.2785
touch+inertia,medium,SE - touch+inertia,medium,W	-0.0281401921	0.006064835	366.01	-4.640	<.0001
midair,short,SE - midair,short,SW	-0.0065820233	0.006064835	366.01	-1.085	0.2785
midair,short,SE - midair,short,W	-0.0281401921	0.006064835	366.01	-4.640	<.0001
touch+inertia,short,SE - touch+inertia,short,SW	-0.0065820233	0.006064835	366.01	-1.085	0.2785
touch+inertia,short,SE - touch+inertia,short,W	-0.0281401921	0.006064835	366.01	-4.640	<.0001
midair,long,SW - midair,long,W	-0.0215581688	0.006064835	366.01	-3.555	0.0004
touch+inertia,long,SW - touch+inertia,long,W	-0.0215581688	0.006064835	366.01	-3.555	0.0004

midair,medium,SW - midair,medium,W	-0.0215581688	0.006064835	366.01	-3.555	0.0004
touch+inertia,medium,SW - touch+inertia,medium,W	-0.0215581688	0.006064835	366.01	-3.555	0.0004
midair,short,SW - midair,short,W	-0.0215581688	0.006064835	366.01	-3.555	0.0004
touch+inertia,short,SW - touch+inertia,short,W	-0.0215581688	0.006064835	366.01	-3.555	0.0004

APPENDIX D

The archive that has been uploaded along with this document contains source files used in the development of the prototype, media resources such as photos and videos and other documents that contain logged data and results from the user questionnaire used in the user study.

The **Archive.zip** file is split into the following folders:

1. **unity-project:** The folder that contains the main source code used for the prototype's development.
2. **media:** This folder stores some photos and videos that have been taken while creating the prototype and during one of the tasks in the experimental phase.
3. **data:** This folder contains the raw data that has been recorded during the user experiments.
4. **data-combiner:** This folder contains the python script necessary for manipulating the raw data logs and preparing them to be used in the data analysis.
5. **evaluation:** This folder contains all evaluation scripts written in R as well as plots and the users' survey results.