Symphony of Modalities for Interaction in Augmented Reality

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

by
Sughosh Krishna Kumar

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr. Ulrik Schroeder

Registration date: 19.01.2016
Submission date: 18.03.2016

Eidesstattliche Versicherung

Sughosh Kumar Krishna Kumar — 327878

Name, Vorname — Matrikelnummer

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

Symphony of Modalities for Interaction in Augmented Reality

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

Aachen, 18.03.2016

Ort, Datum — Unterschrift

Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

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

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

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

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

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Aachen, 18.03.2016

Ort, Datum — Unterschrift

I hereby declare that I have created this work completely on my own and used no other sources or tools than the ones listed, and that I have marked any citations accordingly.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, March 2016
Sughosh Krishna Kumar

Contents

List of Figures

List of Tables

Abstract

The wearable computing space is expanding with smaller and faster technologies. With the addition to these technologies there is a requirement for better and suitable interaction techniques. Interactions between these wearables are also important as they are basis for collocated computing. See-Through Head Worn Display or ST-HWD input techniques are not standardized and rely on proprietary solutions. These solutions include additional input controllers or on-frame devices for interaction having limited interaction space. This thesis is an effort to explore the wearable computing space by designing a Smartwatch as an input controller for ST-HWD.

Larger field of view ST-HWD primarily use cursor techniques to interact. The thesis presents a novel interaction technique called *Symphony of Modalities*, which is a combination of touch and gesture techniques using smartwatch as a controller. Cursor control for large display is designed to achieve coarse pointing through Gesture and fine pointing through Touch.

In a controlled user study, we compared Symphony of modalities with its individual counterparts, Touch, Gesture using smartwatch and also with an input controller supplied by the manufacturer. The study revealed that purely Touch technique was 7% faster than Symphony of modalities, which consecutively was faster than the remaining techniques. However, Symphony of modalities proved to be substantially more accurate compared to all other techniques. Through a questionnaire following the study, Symphony of modalities projected higher average confidence scores and lower average difficult scores, thus acknowledging this technique as an eligible input controller for ST-HWD using cursor based input.

Acknowledgements

I take this opportunity to thank everyone who stood behind me by giving moral support throughout the composition of this thesis.

I am thankful to my mentor and adviser Nur Al-huda Hamdan for her excellent guidance providing me periodically with her valuable insights and constructive feedback which was vital in the formulation and composing of my thesis. I am indebted to Prof. Jan Borchers and Nur for arranging the opportunity to pursue my thesis at the chair and in collaboration with Carl Zeiss.

I extend my gratitude to all my colleagues and friends at Carl Zeiss for their participation in this thesis. I am grateful for my supervisor at Carl Zeiss, Dr. Kai Ströder for his constant support and cooperation in all respects leading to completion of the thesis.

Finally, I thank all my family and friends for their unwavering support.

Thank you!

Conventions

Throughout this thesis we use the following conventions.

Text conventions

Definitions of technical terms or short excursus are set off in colored boxes.

EXCURSUS:
Excursus are detailed discussions of a particular point in a book, usually in an appendix, or digressions in a written text.

Definition: *Excursus*

Source code and implementation symbols are written in typewriter-style text.

```
myClass
```

The whole thesis is written in American English.

Chapter 1

Introduction

"It's very satisfying to take a problem we thought difficult and find a simple solution. The best solutions are always simple."

—Ivan Sutherland

The wearable computing has opened many areas of research in Human Computer Interaction. With Head-worn displays (HWD), Grubert et al. [2015] suggest that the information on HWD is always available in the flick of the eye . Arthur et al. [2015] discuss HWDs were initially designed for military and industrial purpose, however, today it is gearing up for consumer's attention. One of the potential challenges researchers have documented with HWD; is the interaction technique. Techniques such as in-air gesture inputs need to be learned and adapted by first time user, making the *interaction on go* difficult as understood from Piumsomboon et al. [2013]. Notably, the input devices provided by the manufactures of the HWD are not suitable for all purposes.

Smartwatch interactions is an evolving topic of research.

The Smartwatch is a wearable computer that made its debut not too long ago into the consumer industry and is gaining more momentum in HCI research. With the help of the smartwatches, researchers are able to study multiple interaction spaces and metaphors between devices Chen et al. [2014]. The research in collocated devices using the smartwatch has also made many researchers to evaluate smartwatch as a companion input device according to Wu and Yazaki [2015]. The Smartwatches are integrated with sensors that could be used to repurpose it into an input controller.

In this thesis we discuss, how the smartwatch is repurposed as an input controller for a ST-HWD. We introduce a novel interaction technique by combining the existing interactions documented for multiple display sizes. We also describe the process of interaction design and how the smartwatch based input evolved from indirect touch to complete bimanual interaction.

1.1 Motivation

Many advancements in realizing the future of interaction in augmented reality are possible with ST-HMD. These have been successfully studied and documented by researchers. Few researchers recommend gesture based interaction for HWDs Colaço et al. [2013]. The ST-HWDs available today are capable of high real time computations that can be leveraged for vision based hand tracking for in-air gesture input. When compared to touch, gesture techniques are disadvantageous according Moser and Tscheligi [2015]. Potential products such as the GoogleGlass or ODGs R6 have the input interaction space mounted onto the frame of the device. Such techniques increase the access time and the cognitive effort Fukui et al. [2015]. Further, the manufactures of ST-HWD target the ability to mass manufacture and improve the optical design. Here, is an attempt to design a better user experience for such ST-HWDs.

The advantages of smartwatch, evolved as a potential input controller.

The ability of a smartwatch to strap on to one's wrist makes them socially acceptable as they are small and unobtrusive. They are also used for quick and easy access of information in a concise manner. The drawback of the smartwatch is its small display area, both in interaction and information plane. This device is primarily used for reading, dismissing and handling notifications. Further, the information displayed is susceptible for finger occlusions coined as fat finger problem by Siek et al. [2005]. In order to overcome fat finger problem, many researchers over the last few years have proposed powerful techniques such as back of the device (BoD) interaction Baudisch and Chu [2009]. However, existing forms of interaction techniques to overcome this potential problem remains uncertain. One solution that was inspired from our day to day usage of computers was the indirect interaction technique.

Research for smartwatch as input controller is sparse.

The research converged at designing indirect manipulation for the smartwatch display itself Xiao et al. [2014]. The researches were successful in extending the interaction to the watch band Ahn et al. [2015], or on the edges of the smartwatch Houben and Marquardt [2015]. However, the information displayed on a smaller screen always required optimization. The solution to the problem about large information being displayed on a smaller screen was proposed by zooming into the interface in ZoomBoard by Oney et al. [2013]. While doing so we encounter problems such as higher cognitive effort to read smaller text. In addition to cognitive effort, more time is required to zoom into the content. Moving the content onto another device such as smartphone will inherit the problems of the smartphone and higher cost for switching displays Grubert et al. [2015]. A potential solution to these problems encountered would be a ST-HWD. The larger display area facilitates larger content right in front of the eyes while reducing the access time.

The commercial ST-HWD have a FoV of $22°$ to $30°$ translates to a perceived $65 - 85$ feet diagonal display in $16 : 9$ ratio at 3 meters

One of the challenges posed by commercially available ST-HWD, is that virtual content should not be occluded by the physical setup or the environment. This challenge is overcome by an unobtrusive manipulation such as indirect interaction technique, which qualifies to become an eligible candidate. *Indirect interaction* is where an additional input controller. The observable physical interfaces for interaction for the ST-HWD are installed on the frame of the device or with wired controllers. These interfaces for HWD are indirect forms of interactions too but have higher access time and higher cost for switching between the input and output. Further, as in ODGs R6 the interaction area is very small compared to the size of the display.

In-air gestures have several limitations.

In order to overcome the shortcomings in the user experience of such kind, a sustainable solution is required to to be explored. Reitmayr and Schmalstieg [2001] suggest voice and in-air gesture techniques for such devices. Voice or speech based interfaces are not ideal in every scenario. A study from Kollee et al. [2014] suggested that speech based interfaces violates privacy in public spaces and the recognition based errors causes many misinterpretations leading to worse performance compared to other techniques such as gesture. Gesture techniques following vision based tracking have higher computation load and limited to lighting and environmental factors. The lack of haptic feedback and fatigue over period of use, is of major concern in the user experience, as documented in Hincapié-Ramos et al. [2014].

Traditional watches imprint an individual with a kinesthetic memory. Hence, we emphasize the use of proprioception for efficient access and navigation of content on the HWD. Smartwatches support touch and also come integrated with IMUs such as accelerometer, gyroscopes, etc. Using these sensors and the available touch to our advantage, we have engineered a synergy between gesture and touch modalities and have analyzed how this synergy is mapped to the distributed interactions such as the coarse pointing and fine pointing. The advantage of smartwatch as an input controller to a ST-HWD, decouples information and the interaction spaces.

1.2 Thesis Contributions

Contribution is designing and developing of a novel interaction technique.

The contribution of the thesis is three fold. We experimentally determine which watch face yields better proprioception for eyes free interaction.

Second contribution is the designing the smartwatch as an indirect touch controller for a smart glass prototype. We also briefly discuss designing interfaces for this technique.

The primary contribution of the thesis is developing a novel interaction technique **a Symphony of Touch and in-air Gesture** and compare it with its individual input techniques. We conclude our findings with the help of user studies, questionnaires, analysis and retrospection.

1.3 Outline

In this section, we present an outline of the thesis and the ensuing chapters would explain:

Chapter 2 "Related work". This chapter is a discussion of HCI research in the context of this thesis. Existing researches regarding the interactions on a smartwatch and ST-HWD are summarized. We review research on affordances of the watch faces, problems with the existing interaction techniques on smartwatch and ST-HWD.

Chapter 3 "Preliminary Interaction with User Study". This chapter explains and discusses the preliminary user study and preliminary interaction technique. We present an experimental design to evaluate round and rectangular watch-face for better proprioception in order to design eyes-free interaction and develop an interaction technique using indirect touch on a smartwatch. We also briefly explain how an interface on a smartglass is designed to take advantage of the indirect touch.

Chapter 4 "Symphony of Modalities". This chapter revolves around the main contribution of this thesis. We review the drawbacks of preliminary interaction design and explain why and how other techniques would prove useful. We describe the design of the novel interaction technique Symphony of Touch and Gesture. Notably, we introduce the implementation concepts and software prototype to realize the hypotheses.

Chapter 5 "User Study and Evaluation". In this chapter we present the experimental design to study our hypotheses. We illustrate our results and analysis from the user studies. Finally, we also discuss the user‘s preference and behavior through retrospection and questionnaires.

Chapter 6 “Summary and Future Work”. We summarize our findings, methods, principles and present a brief introduction into the possible future work.

Chapter 2

Related work

"Research is formalized curiosity. It is poking and prying with a purpose. "

—Zora Neale Hurston

Smart wearables have provoked researchers to explore and exploit interaction in wearable computing. Smartwatches being more personal are used primarily for health and fitness tracking and faster notification access. The ST-HWD provides powerful means of augmenting virtual spaces and objects in the real world. Applications of augmented reality are also variegated; ranging from health care to process automation. The resulting impact on research by combining the wearable technologies is quite powerful. To harness and explore the effects of such an impact has been the responsibility of the researchers.

2.1 Interaction on ST-HWD

ST-HWD enhances AR experience with digital see-through effect.

ST-HWDs or optical ST-HWDs are wearable devices that have the ability to project virtual information onto the eyes without occluding the view that is currently perceived by the user. ST-HWD have ability to perceive the virtual projection without loosing focus of the real world. It is widely used in Augmented Reality (AR). Grubert et al. [2015] emphasize on a key aspect of higher access time and effort for retrieving smartphones to read information, while with a ST-HWD it is just with a flick of the eyes. Budhiraja et al. [2013a] express that ST-HWD is better suited for AR visualization than any other device because they are always worn in front of the eyes.

The commercially available ST-HWDs are manufactured with integrated or exclusive input devices. These input devices are either wired or mounted on the frame of the glass and prove to have higher access time. The access time is similar to that of retrieving the smartphone. Few ST-HWD supports mid-air gesture by vision based hand tracking. The research from Rekimoto [2001] discuss about requiring high computation for visual based tracking. The major challenge of gesture technique is referred in a survey of Van Krevelen and Poelman [2010] where they discuss how gesture could cause occlusion of the physical environment and hence disturb the user experience in AR. Further the gesture based techniques have substantial effects on performance induced by fatigue Moser and Tscheligi [2015].

Evaluation different input techniques along with their influences.

A good user interaction technique complements a good user experience. Table 2.1 summarizes the existing techniques for ST-HWD. These techniques are conventional and not completely refined keeping in mind variegated use cases. Due to the different environments of use, the user needs to adapt her pattern of usage. This in turn would increase the cognitive effort to work with such devices and consequently may have a contradictory effect on the user experience.

2.2 Interaction on Smartwatch

There exists various interaction techniques for smartwatch.

Smartwatches are designed to be familiar to the existing watch's characteristic such as ubiquity, social acceptability and portability. Boletsis et al. [2015] express that smartwatches could cater to many more uses other than fitness tracking or remote payments as advertised but also in health monitoring. Smartwatches are mainstream comple-

Interaction	Description	Pros	Cons	References
Mid-air Gestures	User arm/wrist movement in 3 dimensional space to perform gestures.	Natural interaction. Clear conceptualization.	Environment dependent. High computational effort. Remembering gestures. Lack of haptic feedback.	Bragdon et al. [2011], Moser and Tscheligi [2015], Piumsomboon et al. [2013]
Finger Controller	Joystick controller with small isometric area to track users' finger to determine acceleration of the pointer.	Portable. Uni-manual and indirect manipulation. Similar to isometric mouse pointer.	Dependent on finger size. Slow and not too precise. Additional hardware.	Douglas and Mithal [1994], Cho et al. [2015] Fukui et al. [2015]
Touch Input	Input controllers such as touch pads, mobile smartphone devices.	Familiar interaction. Flexibility. Support for multi-touch gestures.	Higher access time. Limited maneuverability of the pointers. Higher cognitive effort.	McLaughlin et al. [2009], Serrano et al. [2015]

Table 2.1: Summary of the existing interaction techniques for ST-HWD

mentary devices for smartphones. Hence, they are classified as ubiquitous output devices that are best suited for handling notifications. However, Van Vlaenderen et al. [2015] discuss how smartwatch could be repurposed to be used for handwriting recognition with an in-built camera.

Smartwatches have a limited display size and the interaction and information spaces are the same. Thus, we encounter the documented fat finger problem Siek et al. [2005]. A depiction of this problem is as shown in Figure 2.1 Using the smartwatch‘s display as both interaction and information space is an example for bimanual interaction.

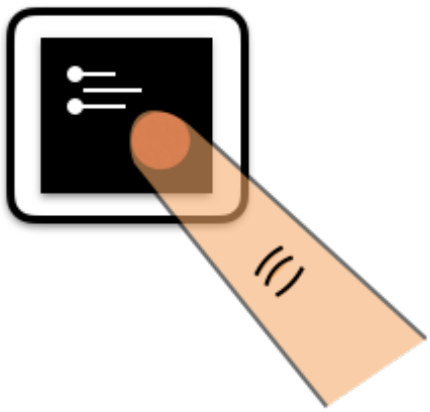


Figure 2.1: Fat finger worries Siek et al. [2005]

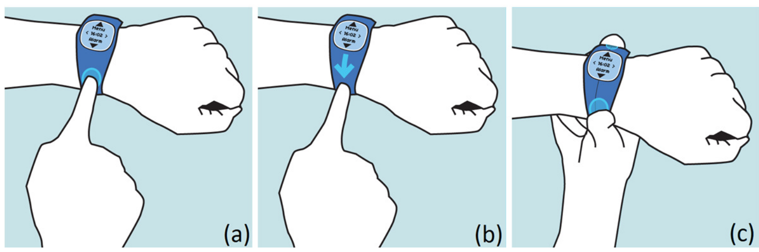


Figure 2.2: WatchIt : Extending interaction to the band shown by Perrault et al. [2013]

A unimanual solution to tackle bimanual interaction problem on smartwatch is gestures.

Smartwatches require two hands for interaction, on one hand the watch is strapped on and other for interacting with the smartwatch. The hand that has the smartwatch strapped is rendered unusable. One solution where just one hand is used to interact with the device was explored in Finger Writing with Smartwatch as shown by Xu et al. [2015]. This research makes use of the inertial measurement units (IMUs) such as accelerometer, gyroscopes to determine arm and finger gestures. They summarized that arm gestures produced highest energy and the finger produced the least energy.

Figure 2.3: SkinButtons Extending interaction space by projecting laser icons. Laput et al. [2014]

2.2.1 Extending Interaction and Information Space

To address the fat finger problem solutions like WatchIt by Perrault et al. [2013] as shown in Figure 2.2, proposed extending the interaction space to the watch band. Another solution proposed was the Skin Buttons Laput et al. [2014] Figure 2.3 that extended the interaction area by tiny laser projected buttons or icons onto the arm. These researches focus on the aspect of retaining the display area and moving the interaction area away. The problem with the display on the smartwatch is that it is too small for larger content.

BandSense Ahn et al. [2015] also address indirect manipulation using band of the smartwatch.

The information space on a smartwatch is generally used to render optimized or small information. A typical example would be text entry on smartwatch. Oney et al. [2013] tried to tackle the larger information on small display with ZoomBoard. The keyboard layout is quite large to fit into the small display making the target sizes for the text buttons very small to touch. ZoomBoard addressed this by providing a zooming capability into a QWERTY keyboard layout and scale them until the buttons are large enough for finger size targets. With this the researchers attained up to 9.3 words per minute.

Combination of input techniques is not new.

Combining input techniques is first seen in *put-that-there* by Bolt [1980]. They use speech and pointing input to interact with graphical user interfaces. Duet by Chen et al. [2014] shows cross device interactions by combining two devices, such as the smartwatch and handheld in order to infer the spatial relationship between the devices and how the interaction could be transferred between the devices as shown in Figure 2.5. MultiFi by Grubert et al. [2015] explore how smartwatch and handheld displays could be extended using a ST-HWD in order to eliminate interaction seams.

2.3 Related Work

Evaluating smartwatch as a potential input controller.

The challenges of the smartwatch could be addressed by moving the information space to ST-HWD and retaining the interaction space on the smartwatch. Thus, the smartwatch could be potentially designed as an indirect input controller for ST-HWD. Smartwatches have evolved from simple digital calculators to modern multi-touch screen with multiple sensors. The earlier linux watch Raghunath and Narayanaswami [2002] proposed constraints for operating systems, interface designs and hardware miniaturization that is possible with today's technology.

Smartwatches have been looked at by researchers as mainstream support devices that are better suited for notification delivery. Since advent of smartwatches they are manufactured in different sizes and faces, primarily the round face and rectangular face. First, it is important to determine which watch face would be best suited for indirect interaction for a rectangular see through display. The parameters under scrutiny to determine the most suitable watch face was

- Better awareness of finger position.
- Easier access to all regions of the screen.
- Ease of eyes free interaction.

Figure 2.4: ARC-Pad (Left) pointer jump with absolute pointing and (Right) relative positioning McCallum and Irani [2009]

Affordances of watch face and its influence.

Ashbrook et al. [2008] suggest that targets around the rim of the watch in case of a round face were easier to target than the targets at the center of the display. The rim was used as a guide to navigate the user's finger towards the target. The authors note that upper left quadrant was likely obscured by the finger and in the bottom area, the targets were difficult to acquire due to the shape of the finger. Blasko and Feiner [2004] discuss how the rims and edges of the watches could be made interactive. The authors also explore the various affordances of the watch face using tactile landmarks. For example, they explain how a physical slider could be integrated into the rim of a round face or physical buttons on the edges of rectangular face for simple menu traversal. Finally they evaluate bidirectional segmented strokes for each watch face. These researches explain how to exploit the watch face for unique interactions based on affordances. However, the research for better watch face for a smartwatch as input is still inconclusive.

2.3.1 Indirect Touch Interaction

Existing research in indirect interaction was reviewed in order to understand as how to engineer the smartwatch as indirect touch input controller. As shown in Figure 2.4 ARC-Pad from McCallum and Irani [2009] explains, how a small mobile screen could be used to control pointer on larger screen. This system is designed with absolute pointing and relative positioning. The touch screen of the mobile device could be used to jump the pointer to a position relative to the screen or move the pointer similar to a touch pad.

Nancel et al. [2013] extend ARC-Pad research by introducing coarse and fine pointing for very large displays. Coarse pointing is similar to the absolute jump in ARC-Pad but the fast-jump traversal of the cursor is based on the control display gain (CD). The fine positioning or slow movement of cursor based on the CD which is in turn based on distance between the targets. The coarse pointing was implemented using a two finger touch on a small area on a tablet and fine positioning by a single finger touch to acquire targets on very large displays. A controlled test by varying control display gain for the input showed that this technique was most accurate. Similarly Gilliot et al. [2014] suggested that users had better proprioception over the edges of the handheld and thus were more accurate for eyes-free input for larger screens.

Indirect Interaction Using a Smartwatch

Indirect touch using smartwatch can be used for interaction.

Since the research regarding the smartwatch as an input controller is limtied, we discuss similar system setups that would aid in design process. Hybrid AR Systems proposed by Budhiraja et al. [2013a] depicts a hybrid of handheld and HWD for AR visualization. Here the HWD is used as a low fidelity output device and handheld device used as an input controller. They discuss how cross device interactions is designed using swipe gestures on the touch screen. Henderson and Feiner [2009] show application of AR using the handheld device as input controller for a ST-HWD for vehicular maintenance. MultiFi by Grubert et al. [2015] also briefly explore the indirect interaction using a smartwatch for a ST-HWD. They summarize that a combination of the ST-HWD and smartwatch can outperform single wearable.

2.3.2 Gesture Based Interaction

Designing of in-air gesture using smartwatch.

Mid-air gestures are quite unique as they represent natural form of interaction. Even though natural, the user experience depends on multiple factors relating to the environment, making it difficult for interaction on the go. Kollee et al. [2014] discuss in-air gesture using vision based tracking achieved using Google Glass for ego-centric gestures. The study concluded that gesture was popular among users compared to voice based interaction.

Today's Smartwatches have quite powerful hardware and also are designed with numerous sensors. With the help of these sensors we could design a gesture recognition system. Porzi et al. [2013] present a smartwatch based gesture recognition system for assisting the visually impaired. Here the authors use different gestures to trigger different functionality and based on critical location, haptic feedback provides a warning to the users. Further, they suggest that the smartwatch themselves are not too powerful, so they use a smartphone paired with the smartwatch to recognize the gesture using a global alignment kernel.

Ambient Interaction by Smartwatches Bieber et al. [2012] suggest how smartwatches could be designed as potential data input devices for head mounted displays. They mention that by using this device for the touch and ability to generate gesture would be computationally efficient for providing input to a HWD. Further, they also discuss the various possible interactions with the smartwatch:

- Click or double click
- Wiping: A back and forth movement without wrist rotation.
- Circle: Circular movement without wrist rotation
- Twist: Rotation of the wrist
- High point: Moving the underarm back and forth.

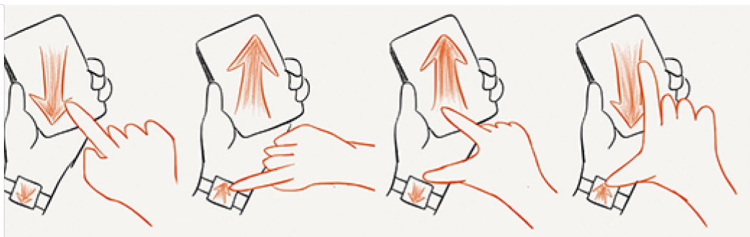


Figure 2.5: Duet: Exploring joint interactions on smartphone and smartwatch Chen et al. [2014]

Challenges of gesture recognition using a smartwatch.

There are several documented limitations of gesture recognition using smartwatches. Smartwatches are ideally worn on the wrist and therefore the elbow/ arm become the controlling joints. The degrees of freedom for the elbow/arm is limited as compared to that of the wrist. The gestures that are generated could be captured using the IMUs but still are susceptible to noise and integration drifts. Bieber et al. [2012] and Xu et al. [2015] discuss why gesture recognition is difficult on a smartwatch. The sensors such as accelerometer or gyroscope alone cannot be used to determine the relative position of the hand in 3D space. The reason being a large drift in observed in sensor values computations over time. To overcome this Bieber et al. [2012] suggest a method popularly known as the sensor fusion. They evaluate the possible gestures with sensor fusion. Further, they have discussed about using multiple sensors available, in synchrony for attaining close to accurate gesture recognition. It was possible to arrive at an approximate relative position of the user's hand and hence detect movement and gestures with this technique.

Collocated device interactions.

The research based on combination of touch and gesture to form a new interaction technique is limited. Research such as Duet Chen et al. [2014] also show how gestures could be made to work with touch synchronously across devices. Knowing the position of the smartwatch worn on the wrist, the authors were able to design a new user experience with cross device and between device interaction. This was first of it's kind where touch and gesture were amalgamated in order to create new interaction metaphors Figure 2.5.

Devices	Portable	Design	Wearable	References
MYO	YES	Except for gesture recognition has no other features. Design resemble a wearable but quite different.	Worn on the upper arm and not too ergonomically comfortable.	Dementyev and Paradiso [2014]
Leap Motion	NO	Has quite a presence and should remain stationary for efficient gesture recognition, should be connected to host device with physical interface.	Not wearable, the interaction on go is very difficult.	Regenbrecht et al. [2013]
Smartphone	YES	Standard design with large touch area and heavy for wearing.	Not wearable without an accessory. Heavy for wearing it on the body	Budhiraja et al. [2013a]
Smartwatch	YES	Lower access time and interaction space is not too small. Better suited for indirect interaction with ST-HWD.	Worn on the upper arm and not too ergonomically comfortable.	Wu and Yazaki [2015],Grubert et al. [2015]

Table 2.2: Review of the potential input controllers

2.3.3 Hardware Decisions for Input Controller

Smartwatch was the hardware of choice for the input controller.

A review of potential input devices was conducted. Table 2.2 summarizes the different input techniques reviewed, their characteristic advantages and disadvantages. The review suggests that smartwatch was most favorable input device complementing aesthetics, usability and social acceptance.

2.4 Summary

We noticed limitations of the interaction techniques in the ST-HWD and smartwatch. Along with solutions that have already been put forth for these problems and their current drawbacks were also discussed. The proposed solution of smartwatch being designed as an input controller for the ST-HWD incurred the challenges that need to be addressed in order to engineer gesture and touch based controller.

Chapter 3

Preliminary Interaction with User Study

"A general principle for all user interface design is to go through all of your design elements and remove them one at a time. If the design works as well without a certain design element, kill it."

—Jakob Nielsen

This chapter introduces the preliminary interaction technique using a smartwatch for a smartglass. This technique is derived from observing results of the user study determined by different smartwatch faces. We discuss about the design process involved in developing this technique along with an interface that exploits this preliminary interaction technique.

3.1 Preliminary User Study

Influence of watch face on positional awareness of the finger can be judged or reported.

The preliminary study was conducted to evaluate the concept in detail of how smartwatch could be repurposed as potential eyes free input controller which relies on the user's proprioception. This study tries to summarize the preference and determination of user for the watch face in eyes free interaction. With the help of retrospection, the watch face with better positional awareness is reported.

3.1.1 Proprioception for Eyes-free Interaction

Influence of positional awareness for eyes-free interaction.

The users have sense of awareness of the watch strapped onto the wrist and they can benefit from this in eyes free interaction. In order to access the smartwatch without glancing at it, boundary conditions of the smartwatch should be determined on the wrist through awareness and proprioception. In touch based eyes-free interactions, the users heavily rely on proprioception. *Proprioception is the unconscious perception of movement caused by body itself.* Human beings have the ability to register the change in position of their body parts based on sensory information.

The smartwatch was favored as the input device as suggested by a review summarized in Table 2.2. It is observable that smartwatch leaves a kinesthetic imprint on the wrist or the wearer. With help of this and proprioception the users are capable of locating the area of interaction on a smartwatch as revealed in Ashbrook et al. [2008]. This user study tries to determine the influence of the shape of the watch face for better awareness in finger position of the user. In addition, observations of various touch based gestures depending on the affordance of the watch face as performed by the users were recorded. Following the study, the decisions based on the watch face were presented.

3.1.2 Study Protocol

Kinesthetic imprint is important to determine location of the watch when eyes are immersed in the content.

Smartwatches are designed similar to traditional watches. The users always tend to have a tactile sense of the watch on their wrist that define the boundary conditions. Our efforts were to analyze, as to what degree the face of the watch itself can influence the awareness of the boundary conditions when strapped on the user‘s wrist.

Commercially available smartwatches come in different shapes and sizes. We compare the round watch face with

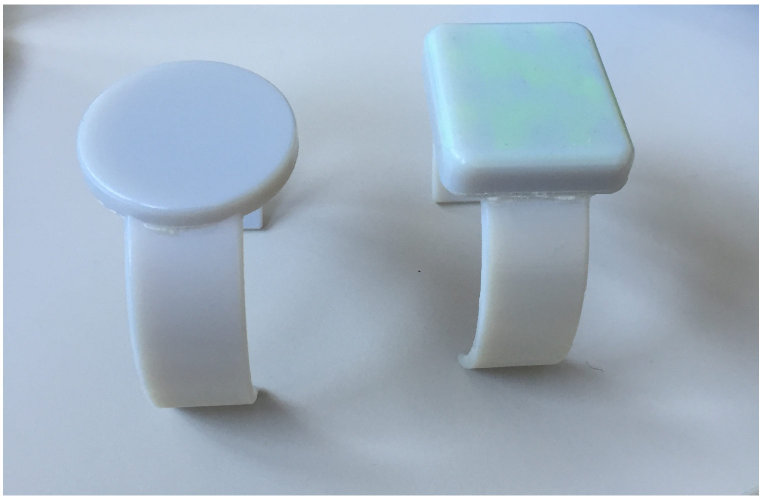


Figure 3.1: 3D printed prototypes used for the test. The dimensions match the commercially available smartwatches.

the rectangular watch face. Through tactile awareness, the users are capable of the determining the boundaries of the smartwatch. The rectangular face has four edges that define the boundary while the round face has no edges but just a rim. Influence of the corresponding face on awareness for users in edge acquisition tasks were studied.

The task was to acquire on all the eight directional edges. Study was conducted with help of two 3D printed smartwatch prototype designs, representing the two faces as shown in Figure 3.1. These design prototypes were modeled after commercially available smartwatch dimensions as outlined in Table 3.1. A display monitor was used to direct the user to acquire the corresponding edge. The experiment was video recorded for evaluation using retrospection.

Directional edges are north, south, east, west, north-east, south-east, north-west and south-west.

Five users participated in a within group study. 3 male and 2 female users, aged between 25 to 35. All users were right handed and were familiar with touch devices. 4 out of 5 users wore watches regularly. The users were seated in front of the display monitor and asked to wear a prototype watch of their preference. A presentation on the

A within group study with 5 participants where users were asked to perform task without looking at the watch.

Objects	Surface Dimension (in mm)	Thickness (in mm)
Rectangular face	42.0×35.9	10.5
Round face	30	10

Table 3.1: Dimensions of the 3D printed prototypes used for the test.

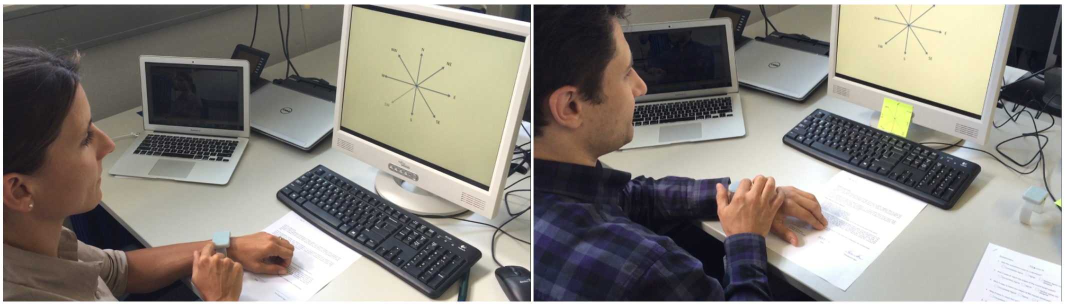


Figure 3.2: Setup for the preliminary study. The users touching the corresponding edge shown on the display. (Left) Rectangular face and (Right) Round face prototypes.

display guided the users to each edge. The users hovered their finger to the desired directional edge and acquire by tapping the edge. For example, if the presentation highlighted the North-East edge, the user had to hover her finger to the North-Eastern extreme edge of the corresponding watch face and tap to acquire. The order in which the edges were displayed were counterbalanced using Latin Squares method in order to prevent carryover effects from one watch face to another. The users were requested not to look at the watch face during the study. Once the presentation for all edges was completed, users were asked to perform a touch gesture they felt natural on the watch face. The experiment was repeated with another watch face. Following the user study, the users were asked to fill a questionnaire and rate each face for each task. Figure 3.2 shows the users performing the edge acquisition tasks.

3.1.3 Results

The results of the study were derived from a post user study questionnaire and retrospective observation of the video recordings. They are summarized with help of two categories. One was the confidence scores and other was the missed edge acquisitions. The questionnaire asked about the user's preference of a watch face in each of the edge acquisition tasks.

Missed edges were analyzed through several video recordings.

Confidence Score

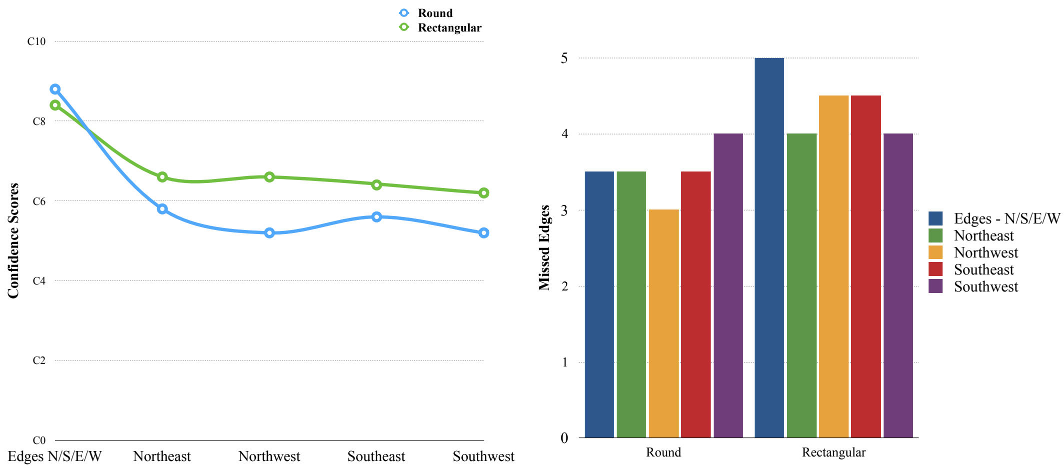
Confidence, in this context was defined as the ability to perceive the exact position of the finger on the corresponding edge of the watch face. The users rated confidence scores for each edge between 0 to 10. The confidence score of 0 denoted no confidence and confidence of 10 denoted extremely confident.

Definition: *Confidence*

During retrospection, only the first trial of edge acquisition was considered. This helped in understanding the immediate response based on kinesthetic imprint and repositioning of the finger after pressure was applied. 3 out of the 5 users preferred the round faced watch. However, their confidence scores and preference for both watch faces was quite contradictory. Users who favored the round face also gave equal or better scores for the rectangular face.

Two users claimed they preferred the round faced watch as they wore a watch having similar watch face every day. One of the users stated *"I felt more confident about my finger position on the round face than on the rectangular face, because I wear a round watch"*. Thus, the user's prior experience with a certain watch face may have influenced the confidence scores. From Figure 3.3(a) depicts the average confidence scores for rectangular and round face respectively. The average confidence score for all users in all directions is $m_{rect} = 6.84$ for the rectangular face opposed to $m_{round} = 5.8$ for the round face. Most participants gave an equal score for more than one edge depending on the watch face. One user expressed that the West region gener-

Rectangular face had better mean confidence scores that round. NW and SW was most difficult.



(a) (b)

Figure 3.3: Graph of (a) mean confidence score for each edges and (b) missed edges on each watch face. The confidence score is between 0 - 10. More missed edges can be observed in round face opposed to rectangular face.

ally felt less confident to reach particularly the North West and South West edges on both the faces.

Missed Edges

Missed edges were wrongly acquired edges on the watch face. For example, if the user was asked to acquire North-east providing a few degrees of error threshold and the user acquired reached towards east this was considered as a missed edge. Through retrospection of the video recordings of the study, many missed edges were recorded on the round face. This was studied as a metric to verify or contradict the user's claim.

Definition: *Missed Edges*

In retrospect, all the users performed better on the rectangular face while at the same time experiencing less missed edges. Even the participants who preferred round face also experienced less missed edges on rectangular face.

Figure 3.3(b) shows the missed edges for the round and rectangular faces. As observed, the users falsely determine their finger position on the round face as there are no edges. The users had problems reaching to even simple directions or non angular directions such as north, east. south and west on the round face accurately. In conclusion, the rectangular watch face was more suitable than round face for eyes-free interaction.

Rectangular face is better suited for eyes-free interaction.

3.1.4 Touch Gestures

Users were asked to visualize the prototype smartwatch as touch device and perform touch gestures that felt natural to them on the watch face. This was studied to determine variations in touch gesture depending on the watch face. It was observed that affordance of watch face influenced the touch gestures.

Touch-gestures were natural to the users.

Affordance of Rectangular Face:Affordance of the rectangular watch face influenced the touch-gestures. These touch-gestures are classified into:

- **Tapping** (similar to tap using a touch screen mobile device)

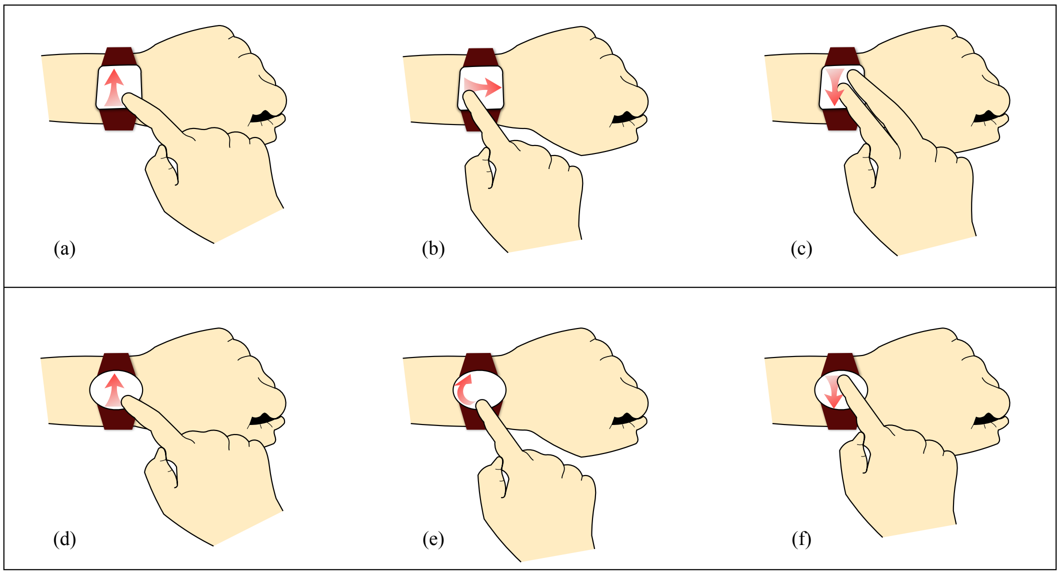


Figure 3.4: Various observed touch gestures based on the affordance of the watch face. (a) Shows flicking, (b) swiping (c) two finger scrolling on rectangular face watch. (d) Shows flicking, (e) sliding along the rim, (f) swiping.

- **Swiping** as discussed by Ashbrook et al. [2008].
- **Flicking / Flinging** is very quick swiping across the screen.
- Two finger scrolling.
- **Pinching** (similar to zooming gesture in touch screen mobile devices)

Affordance of watch face played a vital role in different touch-gestures.

The users were particularly comfortable using swipe gestures from one corner to another irrespective of the direction of the swipe. They also expressed that they found it more intuitive to swipe and flick on the rectangular face.

Affordances of Round Face: The touch gestures based on affordance of the round face had minor differences compared to the rectangular face. These are classified as:

- **Tapping** (similar to tap using a touch screen mobile device)

- **Sliding** along the rim, using the bezel as a guide Froehlich et al. [2007].
- **Flicking / Flinging** is very quick swiping across the screen.
- **Pinching** (similar to zooming gesture in touch screen mobile devices)

The users were comfortable swiping horizontally and vertically through the center. 3 out of 5 participants were capable of performing the sliding along the rim gesture. Figure 3.4 summarizes the different touch gestures based on the watch face.

The study determined the type of watch face yielding better positional awareness and advantages with the finger. Even though few users preferred the round face, their performance on rectangular face was exceptional through better average confidence score than the round face. Further, the touch gestures depended on the shape and affordance of the watch face. However, according to the users, the most intuitive and common amongst these gestures was the flicking gesture. Using the data gathered from this study, the smartwatch was designed as input controller for a HWD with low field of view to control menu based interface.

3.2 Preliminary Interaction Technique

The results of the preliminary user study led to engineering of the smartwatch as an indirect touch input controller . A menu based interface to exploit the designed interaction technique was also developed. This section describes the concepts of indirect touch interaction design and interface design for the smartglass.

Smartglass is the colloquial term for a ST-HWD with data and communication capabilities. The system developed was for the prototype smartglass from Carl Zeiss Smart Optics GmbH.

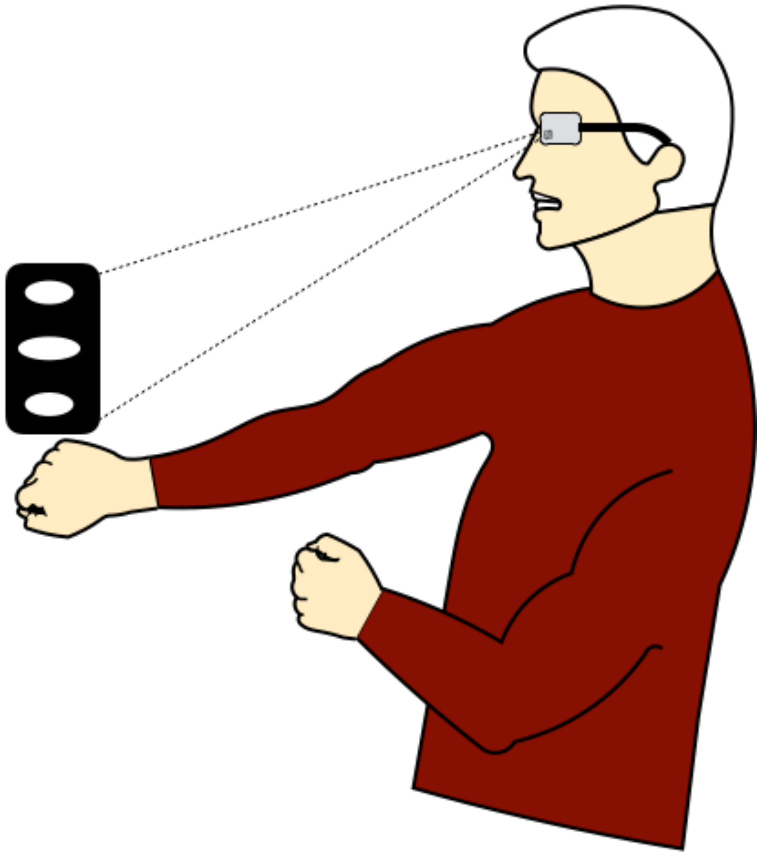


Figure 3.5: Prototype of the optical system displaying content in mid-air at arms length.

3.2.1 Prototype Design

Sony Smartwatch 3 was used as input controller. Display of the smartglass was 5 inch display at 2 feet.

Smartwatch was adapted as the input controller for the smartglass. The compact display size of the smartwatch was not disadvantageous since the display could comfortably accommodate two fingers. Since the information plane was no longer on the smartwatch's display, the information rendered was not occluded by the finger coined as *fat finger* problem by Siek et al. [2005]. Further, the smartwatch could be designed to recognize gestures.

The prototype is a bare essential for showcasing the proof of concept. This prototype system demonstrated the possible user experience. The lens of the smartglass projected a display in mid-air at arms length shown in Figure 3.5. The prototype device ran a Linux based operating system with very small feature set. Only a web browser and a video player were included with the operating system.

3.2.2 Design Principles

Important interaction design principles set forth by Dix [2009] were considered in designing the interaction technique. Using these parameters, design principles for our system were summarized as:

Principles stated are *Learnability*, *Flexibility* and *Robustness*.

1. The interaction should have lower cognitive effort. Interaction should be easy for first time users.

   Easy to understand.

2. The interaction technique should be natural and familiar. The familiarity of the interaction techniques are quite vital in decreasing the cognitive effort Oviatt et al. [2004]. The learning curve for natural interactions remain low Piumsomboon et al. [2013].

   Familiarity with users.

3. Prolonged usage is important. The smartglass themselves would induce a certain amount of noticeable fatigue on the eyes. If the interaction causes further fatigue, it is not suitable for good user experience.

   Sustained endurance to input.

4. Faster and accurate content navigation. Fast access without accuracy is an undesirable combination. Thus, accuracy in content selection is also a key component.

   Accurate and fast.

5. Content occlusion should be avoided. A good practice is where content is not hidden by the physical setup or the environment. In order to achieve better user experience, an interaction technique that would not intrude the field of view of the user perceiving the content is required.

   Avoiding influence of environmental factors.

Gesture technique is not familiar to most of the users and is highly obtrusive. The gesture technique is natural but has a larger learning curve for first time users. As mentioned earlier, gesture technique would induce fatigue over period

In-air gestures failed to comply design principles.

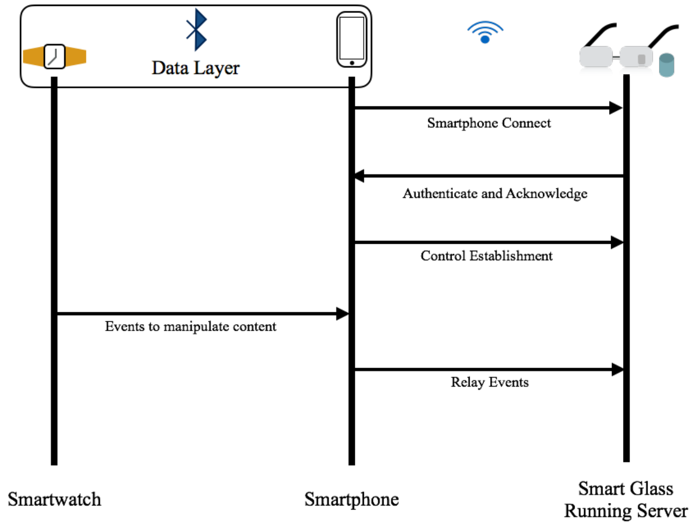


Figure 3.6: Shows timing diagram of the system architecture for the prototype system.

of usage. Lastly, gesture techniques tend to intrude the field of view the user is perceiving. Hence, Gesture based input technique was not considered as a potential candidate for this system.

3.2.3 System Architecture

The components of the integrated system were the smartglass prototype, smartwatch and the smartphone. The smartphone was to relay the smartwatch data onto the smartglass. The architectural decisions depended on the system constraints. The requirement from the manufacturers of the prototype was to run an interactive application on the smartglass using the smartwatch as the controller.

Table 3.2 shows the details of the hardware specification available to develop this interaction technique.

Device	Hardware Specification	Display	Supported Communications
Smartglass	1.2 GHz dual core MIPS system on a chip with a 512 MB RAM	Miro OLED	Bluetooth, WiFi
Smartwatch	1.2 GH ARM A7 processor with 512MB of RAM, 4GB of internal memory and 420 mA battery	1.6 inch, 320x320 pixel resolution Transreflective display.	Bluetooth (DataLayer), WiFi
Smartphone	2.3 GHz processor with 2 GB of RAM and 2300 mA battery	4.95 inch, 1080x1920 pixels HD IPS display.	Bluetooth, WiFi, Cellular

Table 3.2: Hardware specification for the devices used.

The software architecture was designed after various limitations of the system (see Appendix A.2 "Limitations"). The smartwatch was capable of communicating only with the smartphone. Therefore, the smartphone was used to relay the user generated data from the smartwatch to the smartglass prototype using wireless technologies. The software architecture can be summarized in Figure 3.6.

Communication between the smartwatch and smartphone was through data layer.

3.2.4 Interaction Design

In smartglasses, the focus of the user would be predominantly be consumed by the content delivered on the smartglass. Therefore, the interaction must rely on proprioception and kinesthetic imprint of the smartwatch. Smartwatches' are watches with masquerading technology. People wearing a watch, have a kinesthetic imprint of the watch on their forearm. The preliminary study revealed that rectangular face yields better proprioception and tactile awareness of boundary conditions. Therefore, a rectangular smartwatch was used to design this interaction technique.

Focus is consumed by the content on ST-HWD.

Indirect Touch

The priority for design was the user experience and the optical performance of the system. Due to the documented Bluetooth stack software limitation of the devices, all communications between the device were designed using WiFi. Further, the main menu based interface and the corresponding applications were developed on a web browser using HTML, JavaScript and jQuery.

Communication Design

Communication was established using client-server with WiFi.

As the smartwatch is dependent on the smartphone for all of its external communications, an application was developed for the smartwatch to communicate with the smartphone through data layer. The concept and working of the data layer is explained in Appendix A.2 “Bluetooth Stack”.

The operating system on the smartglass proved applications could not utilize the entire infrastructure of the hardware system. Thus, the main interface for the smatglass was developed as a web application.

Smartwatch communicates the event data to smartphone via a protocol.

The smartphone relays the generated data of the smartwatch to the smartglass via WiFi. In order to achieve this, the smartglass was running in a hotspot mode. The smartphone was made to connect to the smartglass’s hotspot and a server would start on the smartglass awaiting the packets from the smartphone. The smartphone meanwhile kept listening to the changes in the data layer induced by the smartwatch. If the smartphone detects changes it forwards the data generated encapsulated into a protocol to the smartglass. The server on the smartglass would interpret and decode this protocol and translate it to input to the XServer (or display server).

The protocols were defined to communicate between the devices. The operating system platforms between the devices were incompatible. The smartglass ran Linux with a gcc compiler and smartwatch and smartphone ran Android with a bytecode compiler. It is analogous to two persons trying to communicate with two different languages. To solve this potential issue we made use of *ProtoBuf*, which is short for protocol buffers. It is an open source platform independent tool that is used to encode data on one platform and decode in other. Using *ProtoBuf* the protocol was encapsulated that contains a scroll value generated from the smartwatch to scroll the web-page on the smartglass.

The efficient method to send the packets across was by designing a UDP (*User Datagram Protocol*) server on the smartglass and a UDP client on the smartphone to send the data across. Both protocols were implemented, but UDP was used over TCP (*Transmission Control Protocol*). The comparison between UDP and TCP for this system can be seen in Appendix A.2 "TCP vs UDP".

3.2.5 Smartwatch as Input Controller

The display of the smartwatch is also the interaction area. This is small around 1.6 inches diagonally but can comfortably accommodate up to two fingers. The smartwatch's ability to read touch input coupled with AndroidWear's APIs for easier access to the read touch data, made it possible to detect input on smartwatch.

The smartwatch application was designed to recognize and send user input to the smartphone. The user input started reading when a user moved her finger on the interaction area. The difference in the distance between current and next position were sent across as scroll value to the smartglass via the smartphone. This proved to be inaccurate, as in one stroke on the interaction surface, hundreds of input values were being generated. The immediate solution applied was to provide timeframe for each value to be recognized every $300ms$. Upon testing with users, this technique also proved inaccurate to navigate the interface. This problem was addressed using Android's GestureDetector library. This library had predefined complex algorithms that detect touch gestures such as *Flinging*. The fling gesture would be generated each time the velocity in the corresponding direction is greater than a velocity threshold.

The fling touch-gesture was used to control menu based interface.

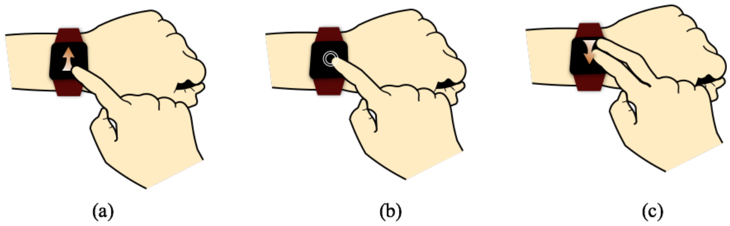


Figure 3.7: Shows the different touch-gesture for interaction. (a) **Flinging** finger to rotate the menu. (b) **Tap** to select the menu item. (c) **Two finger scroll** to stop the video playback.

Each fling gesture generated has an input scroll value of 1. Therefore, the interaction technique using the smartwatch was designed. Further, this considerably reduced the number of input packets that were being sent, making the interaction faster. This technique proved popular with users when the prototype was tested.

3.2.6 Generic Commands

Controls for interaction on smartglass: Fling, Tap, Two Finger Swipe

The summary of the interactions are shown in Figure 3.7. The input to the smartglass was similar to that of a touch device. The fling action triggered the rotation of the menu items on the smartglass in the corresponding direction of the fling. In order to select a menu item the user had to tap the smartwatch screen. Tapping an application triggered an application prototype video. Two finger downward scroll was used to stop the video playback and return to the main menu.

3.3 Interface Design for Smartglass

The prototype was running a constrained operating system limited with various system resources. Due to this limitation, a web application that runs on the browser of the

Figure 3.8: Interface for the smartglass prototype. The main menu is designed to loop and rotates in z axis.

prototype was developed. The prototype depicted potential applications with help of a menu based interface that would trigger prototype videos of applications.

The prototype smartglass supported full color. The specification from the prototype manufacturer demanded that the experience of the user interface follow an approach where the interface is floating in mid-air. In order to accomplish this, the background was chosen to be black. Using black background the foreground becomes conspicuous and appears as floating in a see-through display.

The web application was a vertical circular carousel.

Since the interface provided a feeling of floating in mid-air, a circular menu traversal was designed to show the applications. The circular revolution in z-axis rendered the interface a 3D look and feel. The menu was designed as a circular carousel that would revolve in 3D in a sequence and repeat the sequence following the last menu element. This interface was presented to few controlled set of users and they were quickly able to understand the menu navigation. The display followed a portrait mode resembling a smartphone held at the shortest edge. This display had a non standard resolution of 640×452. Thus, a vertical menu carousel was developed as an interface large enough for a smartphone, meanwhile adhering to the specification of the display resolution. The screenshot of the interface that ran on the prototype is as shown in Figure 3.8. See Appendix A.4 "Iterative Approach" for iterative approach taken towards the interface design and development.

Chapter 4

Symphony of Modalities

"Coming together is a beginning; keeping together is progress; working together is success."

—Henry Ford

In the previous chapter steps to engineer a smartwatch as an input controller for a smart glass with a small FoV was discussed. This chapter explores the various possible interaction techniques inspired to tackle the drawbacks of the previous prototype. The viable techniques for a larger field of view ST-HWD is discussed. These ST-HWDs cater primarily to industrial AR applications that require a FoV of roughly 30°. This is a perceived 65 inches screen viewed from a 10 feet distance.

McLaughlin et al. [2009] suggest with larger displays, the interaction controllers must be capable of faster and intuitive access of all regions of the screen. The prominent solution proposed for interaction by the manufactures of HWD are proprietary or gesture based control.

Large FoV ST-HWD use cursor technique for interaction.

On the other hand, the previous chapters outline documented issues with user interaction with in-air gestures. Budhiraja et al. [2013b] claim *interaction on go* with gesture is cumbersome. One of the important characteristics of the ST-HWD is portability and as previously discussed in Table 2.1, gesture based interfaces are subjected to controlled

Figure 4.1: Showing the *still hand* problem caused by pseudo bi-manual interaction. The hand to which the watch is worn is rendered unusable.

environments making it not ideal for all setups and interaction on go.

4.1 Limitations of Indirect Touch

With indirect touch, *Still Hand* problem is observed.

The indirect touch interaction technique was the first potential solution for designing the smartwatch as an indirect input controller. *Bimanual interaction refers to a system of interaction that requires both hands.* Indirect touch interaction using a smartwatch could be classified as pseudo-bimanual, as there is no complete involvement of the second hand.

Definition: *Still Hand Problem*

STILL HAND PROBLEM:
In smartwatch interaction using touch, the hand to which the watch is strapped is rendered unusable. This problem is coined as the *Still Hand* problem.

Figure 4.2: Commercially available ODG's R6 AR ST-HWDs using small interaction space on frame and an additional ring controller for cursor control.

Smartwatch based indirect touch interaction uses the dominant hand and fingers to interact with the smartwatch display while the hand to which the watch strapped is unusable. Figure 4.1 is a graphical representation of the still hand problem.

The indirect touch is not completely eye-free interaction. During the initial tests, we observed gaze shifts between the smartglass and the smartwatch. However, indirect touch interaction was effectively implemented for menu traversal on the smart glasses with lower field of view.

Devices that have higher FoV, support cursor based interaction as observed in the ODG R6 device. This device supports two controllers, one mounted on the frame of the glass and another as an explicit accessory to control the cursor as shown in Figure 4.2. The visible disadvantage is that both these controllers have small interaction area to control cursor on larger display. Using indirect touch interaction with a smartwatch to control cursor on larger displays has the similar limitations of the small interaction space.

Osterhout Design Group (San Francisco, CA) or ODG is the manufacturer of the ST-HWD and the ring controller.

4.2 Novel Interaction Technique

In order to address the limitations *still hand* problem, limited interaction space, a novel technique was designed. This technique was engineered adhering to the design principles of the interaction techniques as mentioned in 3.2.2 "Design Principles". This interaction technique caters to multiple display sizes using cursor based input.

4.2.1 Design Goals

Identifying an interaction technique with the familiar interaction metaphor of a mouse was beneficial for cursor based control. Therefore, fusing multiple techniques to design a novel interaction technique was the solution.

Using a smartwatch, the right combination of touch and gesture is designed to complement cursor technique.

Existing interaction techniques for HWDs are gesture based or touch based interactions. Indirect touch technique has a benefit of not occluding information on such a setup and being fast Gilliot et al. [2014]. Whereas gesture input technique is more natural Piumsomboon et al. [2013]. Here, we study the effects of combining these two techniques.

The advantage of the smartwatch is that it can be repurposed

As an input controller, smartwatch can be desinged to recognize both touch and in-air gesture. The touch is limited to the display area of the smartwatch. As the watch is strapped to the wrist, the gesture technique cannot benefit from the degrees of freedom (DoF) of the wrist. Gesture techniques instead use the elbow and arm, which requires more work than just wrist movement.

An established uni-manual solution for gesture based interaction with limited movement is twisting the forearm Bieber et al. [2012]. Along with induced fatigue, gesture techniques can also be triggered involuntarily.

Large Display

The ST-HWD with greater FoV was used to implement the novel interaction technique. The observed issue with touch based indirect interaction on a large display is due to difference in the size of the interaction area and the display area. The resolution of input device is much lower to the output display resolution. Therefore, difficulty lies in covering large distances in smaller strokes.

A potential solution discussed by McCallum and Irani [2009] is absolute cursor jump and relative cursor movement. This was achieved using a smartphone as input for 50 inch display. Mimicry of this on the smartwatch is not possible due to the small interaction space. To use smartwatch's touch area, the users must rely on their proprioception and a hunch based on the kinesthetic imprint to perform absolute jumps of the cursor. This is proved to be difficult for most users as discussed in 3.1 "Preliminary User Study".

Coarse and Fine Pointing

Coarse pointing is multiplier of higher CD and fine pointing lower multiple of CD.

Another solution was the coarse and fine pointing as proposed by Nancel et al. [2013]. This technique allowed users to speed the cursor towards the target by jumping the cursor several pixels i.e. the coarse pointing and dynamically adapt the control display gain (CD) near the target as done in Adaptive Pointing by König et al. [2009]. The fine pointing was used to precisely acquire the target. The coarse pointing was implemented with higher CD_{max} and fine pointing with lower CD_{min}.

Designing a state switch between coarse and fine pointing just on the smartwatch's touch interaction space would be a challenge due to the small interaction space. Even though smartwatch can comfortably accommodate two fingers, placing two fingers on the screen, diminishes the area to maneuver. In accordance to this problem, the coarse and fine state switching on using just touch as done was not achievable.

4.2.2 Symphony of Modalities

Gesture:Coarse
Touch:Fine

To overcome the problem of state switching between coarse and fine pointing using the smartwatch led to designing the *symphony of modalities* by combining touch and gesture modalities.

The coarse and fine pointing using the smartwatch was achieved by symphony of modalities. Switching states in symphony was designed

- The coarse pointing by the gesture technique.
- The fine pointing by the touch technique.

4.3 Design Process

A powerful hardware and complementing software was used to study symphony.

A system to test the symphony was developed after evaluating hardwares and softwares specifications carefully.The early prototype device ran a constrained operating system with access to only the web browser with a small FoV. This was not considered to study effects of symphony of modalities. Powerful software system and complementing hardware required to analyze the this technique was reviewed.

4.3.1 Hardware Decisions

ODG R6 and Sony Smartwatch 3 were chosen for this study.

To study the symphony of modalities, larger FoV device was required. Larger devices such as the ODG R6 had adequate processing power to be used to explore this technique. This device supported cursor technique.

The initial exploration of the modality determined suitability in cursor based tasks in larger FoV. The specification ODG R6 shown in Figure 4.3(b). The smartwatch chosen was compatible in software version with the ODG R6. The hardware of the smartwatch was identical to the initial prototype discussed previously shown in Figure 4.3(a).

4.4 Interaction Design

Touch and gesture are the widely used interaction techniques in HWDs. Symphony of touch and gesture is an

(a) (b)

Figure 4.3: Shows hardware used to build and test the symphony of modalities interaction technique. Smartwatch is used to develop the technique and ODG R6 is used to test the technique, both running on Android.

effective combination of two individual interactive techniques. Touch interactions on the display area of the smartwatch and gesture control using the sensors of smartwatch. Testing for synergy between touch and gesture would prove the symphony of modalities as better input technique for cursor control.

4.4.1 Control Display Ratio

Control display ratio (CDR) was used to design the coarse and fine pointing as opposed to CD by Nancel et al. [2013]. CDR are used widely is cursor acceleration where input and output plane sizes are known Blanch et al. [2004]. The CDR is the ratio of resolution of output device to the input device given in Equation 4.1:

CDR was used to design coarse and fine pointing.

$$CDR_x = \frac{ScreenWidth}{InteractionWidth}$$
$$CDR_y = \frac{ScreenHeight}{InteractionHeight} \tag{4.1}$$

The interaction width and height of the smartwatch is similar to its display size. Angular rotation was used to recognize gesture, therefore, the interaction width and height are the complete possible degrees of movement i.e. $360°$.

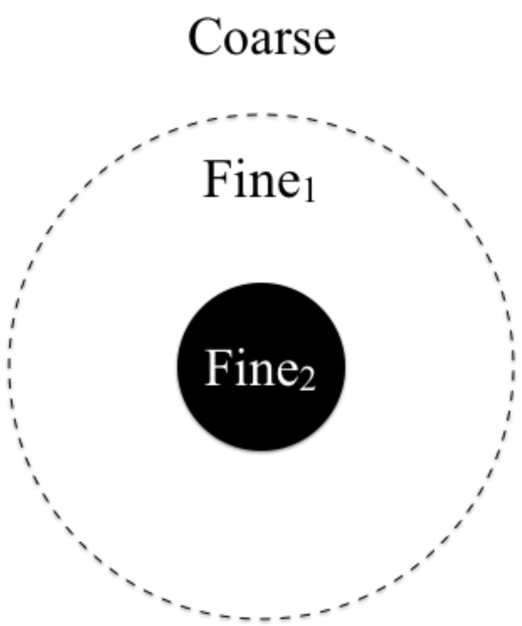


Figure 4.4: Showing the coarse and fine pointing. Black circle denotes the target. Fine_1 is $3\times$radius of target and Fine_2 is the target area itself.

There were two levels for fine pointing.

The fine pointing region was defined by the target width. The cursor remained in coarse pointing mode until it entered the fine pointing area. There were two levels of fine pointing area for the target.

1. The cursor was said to be in first fine control level when it was in area of $3 \times W$ around the target where W is the target width from the target's center.
2. The second fine control level was when the cursor entered the target area itself.

The coarse and fine pointing regions are summarized in Figure 4.4.

4.4.2 Touch Interaction

Touch interaction is gaining familiarity with users of all ages. Touch interaction is engineered using the display area of the smartwatch which is also the interaction area on the smartwatch. Indirect touch input technique to control cursor is similar to a touch pad on a laptop. However, in this indirect touch interaction using a smartwatch, the interaction space is available on the user's body. The user's finger position on this area is read as current position and the next position. In order to determine the pointer movement in a particular axis, the difference between current axis value and the next axis value is computed and sent across to ST-HWD to control the cursor in the desired direction.

The standard axis x-axis is in direction of the crown and y axis is in the direction perpendicular to the crown.

	Fine$_1$	**Fine$_2$**	**Coarse**
Touch	$0.3\times CDR$	$0.3\times CDR$	$1\times CDR$
Gesture	$0.3\times CDR$	$0.3\times CDR$	$1\times CDR$
Symphony	$0.5\times CDR$	$0.2\times CDR$	$1.5\times CDR$

Table 4.1: Fine and coarse pointing through CDR multipliers.

With *CDR* using the smartwatch as touch input for ST-HWD, the users could move the pointer half the width of the screen in one edge to edge swipe on the smartwatch. Cursor control was faster in this mode due to dynamic adaptation of CDR. The summary of CDR multiplied with constant values to enhance cursor control in coarse and fine pointing is in Table 4.1.

CDR for touch was ScreenSize $\div$ Smartwatch-Display Size in pixels.

This technique felt natural to the users familiar with touch devices. Indirect touch using smartwatch and the corresponding axes is shown in Figure 4.5(a). Finally, the users had to tap on interaction space to register a selection. This was confirmed by a vibration feedback from the ST-HWD.

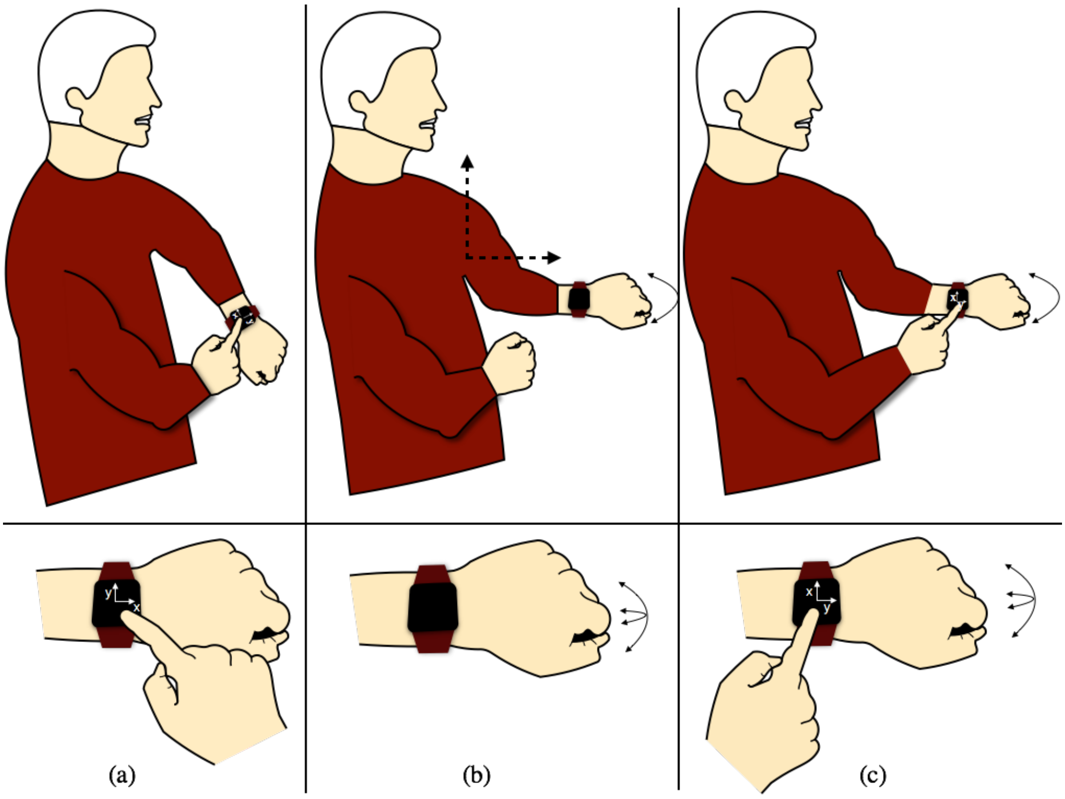


Figure 4.5: Shows the various types of interactions designed using the smartwatch. (a) Shows the TOUCH behavior where user folds the arms inwards to interact with the dominant hand. (b) Depicts the GESTURE technique where users' arms had to be perpendicular to the body. (c) SYMPHONY technique uses both touch and gesture where touch has its axes inverted due to prescribed position of the arm and the approach direction of the finger.

4.4.3 Gesture Recognition

Gesture interaction has been popular in AR research for a long time. Gesture technique using a smartwatch is designed using the sensory information. The smartwatch has integrated sensors such as accelerometer, gyroscopes and magnetometer that are leveraged to construct a gesture recognizer.

Purely gyroscopic data is susceptible to gyro drift.

With purely accelerometer data, it is difficult to recognize gesture. Determining the user's arm position based on acceleration data in the direction of gravity is susceptible to large integration drifts. (See Appendix A.2 "Gesture Recognition" for detailed limitations of gesture using just accelerometer.)

Gyroscope is used to determine orientation changes in the device. It is far more accurate and has a short response time. The disadvantage of gyroscope is the gyro drift. The gyroscope provides the angular rotation speeds for all three axes. To get the actual orientation those speed values need to be integrated over time. This is accomplished by multiplying the angular speeds with the time interval from the last and the current sensor output. However, the limitations of integration drifts make the orientation values imprecise. The gesture technique posture is as shown in Figure 4.5(b).

Sensor Fusion

Sensor fusion is an existing solution for gesture recognition.

Sensor Fusion is a technique in which multiple sensory data are combined to construct a system. This technique is used to avoid both, gyro drift and noisy orientation. The gyroscope output is applied only for orientation changes in short time intervals, while the magnetometer and accelerometer data is used as support information over long periods of time. Sensor fusion is equivalent to low-pass filtering of the accelerometer and magnetic field sensor signals and high-pass filtering of the gyroscope signals. (See Appendix A.4 "Sensor Fusion")

Sensor fusion is currently the best method put forward by researchers to overcome the gesture recognition problem using smartwatch Bieber et al. [2012]. Drawbacks still exists as the device is very sensitive to orientation change and as user's arm is bent inwards the orientation keeps changing. This made the cursor behave unexpectedly. Therefore, a fixed posture to perform gesture is required. The posture dictates that the arm performing gesture be perpendicular to the body and not bent inwards. The users had to tap the screen of the smartwatch to confirm selection.

CDR was calculated as the ratio of the screen size to $360°$ of input. The angular displacement was calculated by summing the previous data. Further, the low pass filter used cutoff noisy rotational values. Therefore, in order to move the cursor 3.5 pixels in y-axis the user had to move 10mm in the corresponding angular direction. In x-axis the user had to move her arm by 10mm to move the cursor by 2 pixels. The multipliers for coarse and fine pointing mode are show in Table 4.1.

CDR for gesture was ScreenSize$\div 360°$. 10mm arm movement was equal to 3.5 pixels cursor movement

4.4.4 Symphony of Touch and Gesture

Symphony of modalities is a technique where gesture technique is used for coarse pointing and touch is used for fine pointing where gesture recognition is developed using sensor fusion. The control display ratio for gesture input is greater than 1 for the pointer to jump several pixels in a single movement. The pointer slowed near the target by dynamic adaptation of control display ratio to facilitate user to switch modes to touch.

Indirect touch interaction read by the smartwatch is used to perform fine or precise pointing. The *CDR* was dynamically adapted by decelerating the pointer as it neared the center of the target for precise acquisition.

As the arm posture was recommended to be perpendicular to the body, the x and y axes on touch area were inverted from the standard touch as depicted in Figure 4.5(c). The standard axis model for touch followed, x-axis in the direction of the crown and y-axis being perpendicular to the crown. Opposed to this, in symphony, x-axis was perpendicular to the crown and y-axis was in direction of the crown. When the orientation of the watch changed by greater than $100°$ i.e. if the arms were bent completely inward, the axes were switched dynamically to the standard touch model as in Figure 4.5(a).

Touch mapping was inverted due to prescribed arm posture and approach direction of interaction finger.

Clutching

Clutching is defined as halting the movement of pointer in order to reposition the arm when maximum movement of the arm has reached. This was important in gesture technique using the smartwatch in order to avoid erratic cursor movement induced by fatigue of arm movement. Clutching gave the ability for users to reposition their arms without intervening in the cursor progress. Clutching was implemented with help of the touch area of the smartwatch. This was originally designed for the gesture mode. While in gesture mode, if the user performed a touch operation on the smartwatch the cursor movement stopped and the user could reposition her arm accordingly.

Clutching was used in gesture mode to reposition the arm without disturbing the current cursor's position.

Clutching was a product of symphony. This feature stopped the coarse pointer movement due to gesture when the user switched to fine pointing using touch. Hence, the cursor was not influenced by movement of arm during fine pointing, making acquisition was more precise.

The posture for gesture in symphony of modalities was same as in gesture technique i.e. arm's position was perpendicular to the body. The technique is depicted in Figure 4.5(c). When the user had to select a target, she had to tap on the touch screen of the smartwatch.

The *CDR* for symphony was different for coarse and fine pointing. As gesture technique was used for coarse pointing the multiplier constant value was > 1 and CDR for fine pointing was determined by CDR of touch multiplying with the different constant value < 1 at each level. The values chosen for fine and coarse pointing for symphony are outlined in Table 4.1. A constant multiplier 1.5 for coarse pointing is to exemplify single movement of 10mm to move the cursor by 5 pixels as opposed to 3 pixels in standard gesture. The fine_1 was chosen to be higher than fine_2 as the cursor deceleration further aids precision in pointing.

CDR for gesture was for coarse pointing and CDR for touch pointing was fine pointing.

4.5 Software Design to Study Symphony

The symphony of the modalities had to be studied with help of an application. The architecture of the application was defined to explore the different states and the predicted behavior of the input. An application on the smartwatch recognized both touch and gesture and used the persistent class to encapsulate data and send it to the ST-HWD. Another application on the ST-HWD recognized the input from the smartwatch, distinguished between touch and gesture states and move the pointer in desired direction. This application's interface was a circular target Fitts Law task MacKenzie [1992].

4.5.1 Challenges and Solutions

Later version of AndroidWear made it possible for stand-alone communication of smartwatch using WiFi.

The challenges were addressed before the development process. The ST-HWD ran Android 4.4 an older version of the Android operating system optimized for stereoscopic see through. Thus, the communication between smartwatch and ST-HWD using bluetooth was not possible. This constraint exists as the earliest Android OS capable of Bluetooth communication with smartwatch was 4.4.2 version. Changing the version of the operating system by using third party builds would cause the stereoscopic view to malfunction.

The latest update to AndroidWear 5.1 provided direct access to WiFi on the smartwatch, which in the previous versions, used the paired smartphone to connect through WiFi. Based on software decisions, the ST-HWD was in hotspot mode and the smartwatch connected to this hotspot to communicate the input through UDP.

Injecting events on ODG R6 was not possible.

The cursor control on the ODG R6 ST-HWD was locked by the manufacturer. Injecting events into the device using system classes did not manipulate the cursor. Therefore, a dedicated cursor technique that responds to the input from the smartwatch was developed. The cursor was part of the Fitts Law application.

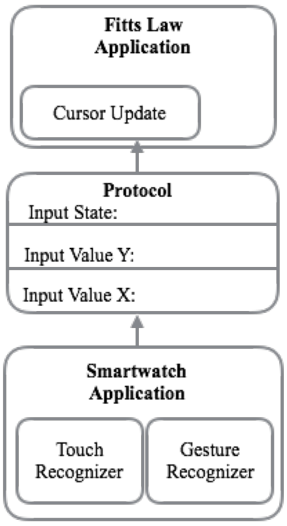


Figure 4.6: Shows the architecture of the of the abstract system using a defined protocol.

4.5.2 Software Architecture

The system architecture can be classified by input and output devices. The input was generated using the smartwatch and the output display was the ST-HWD. The application on ST-HWD reacted to the smartwatch input.

The pointer was designed to resemble a circular bull's eye. Colors used for the pointer was conspicuous and user could distinguish pointer clearly from the targets or background. The application followed client-server model. Server running on the ST-HWD was sensitive to the smartwatch client generated input.

Protocol: X cursor value, Y cursor value and an integer value indicating touch or gesture.

The communication between ST-HWD and smartwatch required a protocol. This protocol definition consisted of the data that manipulates the pointer. This consisted of state of the input, value in x axis and value in y axis. The complete architecture with the protocol definition is as in Figure 4.6.

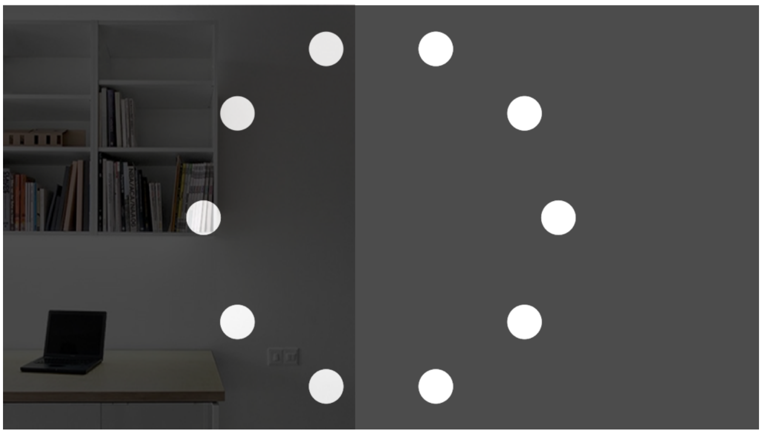


Figure 4.7: See-through application showing 10 targets to acquire of equal width placed at equal distance.

4.5.3 Software Development

2 major components: Touch Recognizer and Gesture Recognizer.

The smartwatch application had three software components they are a touch recognizer, a gesture recognizer and UDP client. The gesture recognizer sent updates of the orientation shift. The axis of touch was dynamically adapted when the orientation shift was greater than 100°.

Touch recognizer computed the cursor movement by the difference between the previous and the next position on the smartwatch touch/display area.

Gesture recognizer through sensor fusion recognized gestures. This was also used to determine the orientation shift caused due to posture of arm at realtime.

UDP client was used to communicate the input data from the smartwatch to the ST-HWD. The application on the ST-HWD had four components a Fitts Layout, Cursor Controller, UDP server and Data Generator. Fitts Layout depicted 10 circular target acquisition Fitts law task as shown in Figure 4.7. This view was updated each time the cursor position was changed. When the user moved the cursor on top of the target, the color of the target changed to provide visual feedback. Upon acquisition of the target a discrete vibration feedback from the HWD was provided.

The Cursor Controller component controlled the cursor movement when data from client was injected. This module was responsible to determine the cursor behavior, corrections of control display ratio, switching between states of coarse and fine pointing.

The UDP server was responsible to read and decode input data sent from the smartwatch.

The Data Generator encapsulated the user's behavior with the input controller and corresponding technique in XML. The encapsulated data was analyzed after experiment to infer results of the hypotheses. Further details about the software development such as algorithms and class structures are discussed in Appendix A.

Chapter 5

User Study and Evaluation

"To me, error analysis is sweet spot for improvement."

—Donald Norman

Previous chapters discuss about the limitations in interaction techniques that paved path for the to symphony of modalites by combing touch with gesture to perform fast and accurate cursor based tasks. This chapter discusses the various steps involved in verifying this technique by an experiment and empirical validation.

5.1 Prelude

Aim: Evaluate symphony of modalities with its individual counter parts and manufacturer supplied controller.

The aim of the user study was to evaluate the symphony of modalities by comparing it to purely touch and purely gesture technique. Further, this study is used to investigate the validity of this technique with the manufacturer supplied input controller called ring/base controller as the baseline for comparison. With help of the study we analyze and evaluate different properties of the techniques such as accuracy, task completion time (speed), throughput and fine control of the technique. The final hypothesis was evolved from multiple research questions.

5.1.1 Research Questions and Hypotheses

The basis of research is to tackle a problem or a question by formulating hypotheses. A basic research question initiated the topic of discussion that was further explored through experimentation and further questioning. The preliminary interaction design was developed after determining the suitable watch face. To accomplish this, the research question and the hypotheses put forth was:

How to repurpose a smartwatch an input controller for ST-HWD?.

- **H$_0$**: The watch face does not influence awareness and proprioception for eyes-free interaction.
- **H$_1$**: The rectangular watch face has influences awareness and proprioception more than the round face watch for eyes-free interaction.

RQ: *Which watch face is yields better proprioception for eye-free interaction with ST-HWD?*

The null hypothesis (H$_0$) was refuted and it was observed that rectangular face yielded better awareness and proprioception for eyes-free interaction. From this study, preliminary interaction technique was developed. The limitations of this technique and requirement to address larger displays led to the development of symphony of modalities. However, the question remained:

RQ: *How can a combination of touch and gesture modalities be used to manipulate cursor technique for object acquisition in AR interfaces?*

- **H$_{01}$**: Symphony of modalities is not different from touch, gesture or native controller in terms
  - *Precision of the pointing.*
  - *Speed in target acquisition tasks.*
  - *Throughput.*
  - *Fine control.*
- **H$_1$**: Symphony of touch and gesture allows more
  - *Precision of the pointing.*
  - *Speed in target acquisition tasks.*

RQ: *Would the symphony of modalities prove to be more efficient than its individual counterparts and manufacturer supplied controller?*

Figure 5.1: Application layout for Fitts law from a see-through perspective. Yellow denotes the current target and red is the next target.

- *Throughput.*
- *Fine control.*

as compared to touch, gesture and the native controller.

- **H_{02}**: Size of the targets has no influence on the task completion time.
- **H_2**: The size of targets influences overall task completion time.

H_{01} and H_{02} are null hypotheses formulated from the research questions.

Symphony of modalities combined gesture for coarse pointing and touch for fine pointing. In this technique, users could engage both hands in interaction. Further, changing the size of the target would increase or decrease difficulty in the pointing task. Thus, the hypothesis H_{02} and H_2 also was of interest.

A within-group experiment was designed in order to study the aforementioned hypotheses. This was an empirical study that used generated data to analyze and report the findings. The task was a simple Fitts Law as by MacKenzie [1992]. An overview of Fitts Law could be seen in Appendix A.1 "Fitts Law". The experiment layout followed 10 circular targets with a fixed distance D between the targets. The targets were layed out in a circular pattern as shown in Figure 5.1.

5.1.2 Independent and Dependent Variables

The independent variables that were varied to observe corresponding dependent variables in the experiment.

Dependent Variables

Dependent variables along with their operational definition according to the hypotheses are:

- *Movement time* measured in milliseconds, denotes the time taken by the user to move the cursor over to the target and acquire. Scale: Interval
- *Error* measured in pixels is the distance between the center of the cursor and center of the target at acquisition. Scale: Interval
- *Overshooting* measured as a logical value with two levels, *Yes* or *No*. Yes stating overshoot occurred and No denoting no overshoot. Scale: Ordinal
- *Throughput* measured in bits/millisecond shows the throughput of the technique. Scale: Interval

Independent Variables

The independent variables and their method of manipulation are:

- *Interaction Technique*: This variable consisted of 4 levels denoting individual modalities. Scale: Nominal
    1. Touch

2. Gesture
3. Symphony of Modalities
4. Base Controller

The data was for each dependent variable for each technique was recorded.

Scale: Interval

- *Index of Difficulty* (ID): This is a Fitts law parameter measured in bits that influences the size of the targets. There were 3 levels for this independent variable:

From Fitts law:

$$ID = log_2(\frac{D}{W} + 1)$$

where D is constant and W is defined through ID.

1. 3.5
2. 4.0
3. 4.5

Higher values denote smaller target sizes and lower values denote larger targets. The manipulation of these targets were counterbalanced using Latin square.

5.2 Task Design

Pilot study revealed maximum distance possible between targets is 600px.

The experiment was a Fitts law task in which every participant had to acquire the targets using four different interaction techniques. This was designed to evaluate the most user friendly technique among them. This Fitts law task was defined by fixed distance between the targets *D*. This distance was set to a constant value as it aided the extreme cursor movement in all directions. The index of difficulty (*ID*) were carefully chosen after several trials. The summary of the parameters chosen for the Fitts Task is in Table 5.1.

Parameter	Description	Value	Unit
D	Distance	600	Pixels
ID	Index of Difficulty	3.5 4.0 4.5	Bits
W	Width	58.175 40.0 27.743	Pixels

Table 5.1: Fitts law task parameters and values

The target acquisition task began by asking the participants to wear the smartwatch and the ST-HWD that was running the Fitts law task.The participants then had to move the cursor visible through the HWD to the desired target's center position and acquire the target. A vibration feedback along with a visual feedback on the ST-HWD was provided to confirm the target acquisition. This tasks remained the same across modalities.

In pilot study we found the lowest threshold for of target size corresponds to ID 4.5.

The participants were asked to complete the task as fast and accurate as possible. It was also emphasized that the higher accuracy means acquiring target at its center. The tasks designed was up to 90 minutes that included several repetitions. The data of the cursor movement, time required for target acquisition and the precise position of the cursor at acquisition is recorded. Following the task the users were asked to fill out a questionnaire and share their experience about the interaction technique with the researcher.

The path data along with the task completion time data were logged. Implication of this data is discussed in 5.6 "Evaluation".

5.3 Participants

Twelve participants took part in the study, 7 female users and 5 male users. The demography of the participants ranged from bachelor students interns to scientists with

This range of ages were considered to generalize user's preference with a technique.

several years of experience. These participants were aged between 20 to 46 years with a median of $median_{age} = 25.5$. All participants were right handed except one participant and 9 of the 12 participants had corrected sight. The users with corrected vision, except one user wore contact lenses. Therefore, there were no complaints about visual disturbances caused due to incorrect vision or by wearing spectacles along with the ST-HWD. All participants were familiar with touch based interaction and only a couple were familiar with gesture interaction but none of the them had previously worked on a ST-HWD.

5.4 Experimental Design

The first of the 10 targets in the acquisition task was not considered.

The experiment designed to study user experience had 10 circular targets to acquire. The first target was not considered due to ambiguous starting position of cursor relative to other trials. Next target was displayed after successful acquisition of the current target. This experiment was divided into 3 blocks. Each block had the 3 chosen index of difficulty discussed earlier in Table 5.1 which translates to 3 different widths. The *ID*s in each block were counterbalanced using Latin squares to avoid carryover effects. The participants had to perform this task on each block with 4 different techniques:

1. **Touch**: Smartwatch was strapped on to the wrist and using indirect touch interaction, the cursor on HWD was controlled.
2. **Gesture**: Smartwatch was strapped on to the wrist and when users moved their arm the cursor on HWD moved.
3. **Symphony**: Smartwatch was strapped on to the wrist and moving the arm triggered gesture based coarse jump and touch for fine pointing in the vicinity of the target.
4. **Base Controller**: The isometric joystick that is worn on a finger to control the cursor and select the target.

(a) (b) (c)

Figure 5.2: Apparatus used in performing the experiment showing (a) ODG R6 with ring/base controller, (b) Sony Smartwatch 3 and (c) Macbook Pro 15″

Before beginning every experimental block with each technique the user was made to practice the task with respective technique. Upon the user confirming that she was confident to begin the task the experiment began.

Users were insisted to practice the interaction technique before the task.

The participants had to complete both touch and gesture tasks successfully before they could begin with the symphony. This was done so as to familiarize the participants with the individual techniques to avoid confounding effects.

Each block consisted of 10 targets across 3 different *ID* values that alter the width (*W*) of the target. To avoid learning effects from one trial to another, the *ID* values in each block were counterbalanced. Each block was repeated 3 times to obtain quantifiable data.

Total number of trials per user: $3 \times 3 \times 4 \times 9 = 324$.

The number of users tested were 12. Therefore, the total number of trials recoded for evaluation are $12_{\text{USERS}} \times 9_{\text{TARGETS}} \times 3_{\text{BLOCKS}} \times 3_{\text{ID}} \times 4_{\text{TECHNIQUES}} = 3888$.

5.4.1 Apparatus

The experiment was conducted with the apparatus summarized in Figure 5.2. The ODG R6 optical see through HWD was powered by Texas OMAP SOC with bluetooth and WiFi antennae. This ST-HWD ran the Fitts Law application. An additional finger wearable controller called the base controller that resembled an isometric joystick sup-

Smartwatch 3 had display/interaction area of 320×320 pixels and supported WiFi. ODGs display resolution was 1280×645 pixels with $30°$ FoV.

plied by the manufactures of ODG R6 was also used for cursor control. Sony Smartwatch 3 with the integrated sensors such as accelerometer, gyroscope and magnetometer was used as input controller. Post experiment data analysis and the switching between the techniques was accomplished with help of a Macbook Pro 15″ with Java support.

5.5 Procedure

The experiments were video recorded with user's consent. The overall study lasted 90 minutes and the user could pause any point in time. If the user paused during a task, she was asked to repeat that particular block again.

Users consented to be recorded over the video for retrospection.

Before beginning the experiment, the participants were requested to fill out a consent form, furnish demographic background information along with their previous experience with touch and gesture techniques and their experience working on HWDs. They were provided with a brief description about the study and tasks they were about to embark. Then, they were requested to wear the HWDs and the Smartwatch. At the beginning of each task, the participants were provided a brief introduction to the interaction technique.

The participants were asked to perform trials until they felt confident to begin the task with a particular modality. Each task was repeated 3 times to obtain quantifiable data for later observation. Every pattern of movements of the cursor was recorded during the process. Following the user study, the users were asked to fill out a questionnaire (see Appendix B) and take part in a semi-structured interview to share their experience of each of the techniques they had tested.

5.6 Evaluation

Evaluation of the technique was accomplished based on the observations. The observations were from four different sources: generated data from the application, questionnaires, demographic information and video recordings.

5.6.1 Path Analysis

The user generated data from the application is used for path analysis. This was a structured XML data that consisted of every target position in 2D, corresponding cursor's position on every movement in 2D, relative time and absolute time for cursor traversal. In order to analyze the data, a JavaFX application was developed that would reconstruct the application layout and path patterns of each trial. The analysis of each of the task:

Touch technique had steps in cursor movement.

- **Touch**: In this task the cursor movement for most part was as predicted. It followed a smooth trajectory and acquisition was deviated from the center by few pixels. In few cases the path was closer to the distance between the targets. Straight path line was rare, however, a step pattern could be noticed. Figure 5.3(a) shows the path patterns for touch.

Gesture technique was both difficult to use and difficult to analyze

- **Gesture**: The cursor movement was found to be erratic and lack of consistency was visible. There were unforeseen jumps from one position to another during the cursor movement. This was caused due to changing the orientation of the watch rapidly from the prescribed position. Further, patterns revealed a difficulty in positioning the cursor in target's center for acquisition showing a lack of fine control. The total travel distance was very high in some cases more than twice the distance between targets. Figure 5.3(b) depicts path patterns for gesture technique.
- **Symphony**: The patterns observed in SYMPHONY were clear and revealed better consistent movement

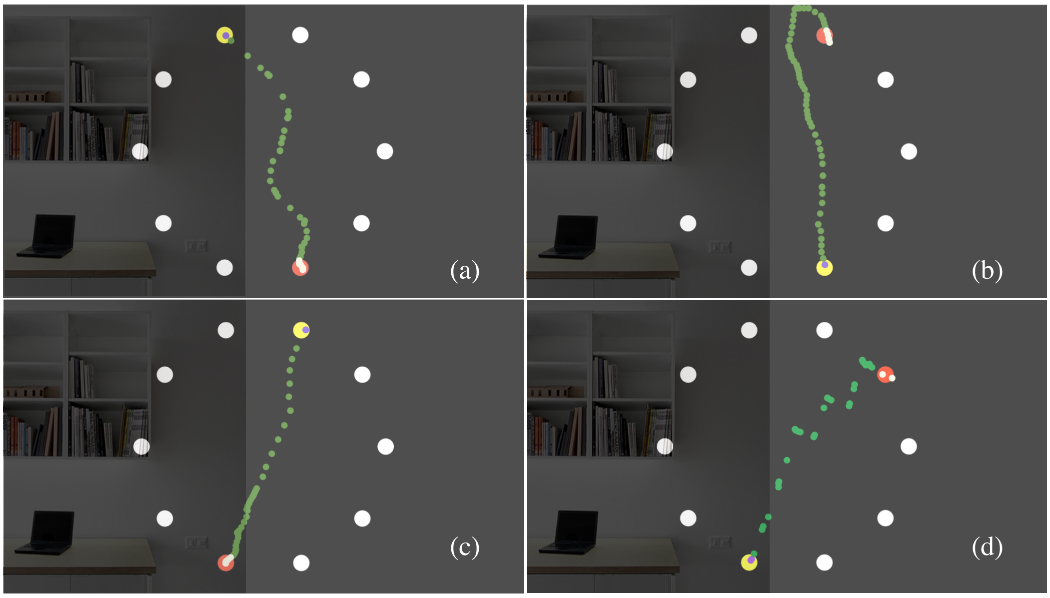


Figure 5.3: Path patterns for each interaction technique. (a) Touch: Pattern is not straight and the acquisition is not at the center. (b) Gesture: The fine control suffers from inaccurate traversal of cursor. (c) Symphony: Patterns follows a straight path and close to the distance between targets. (d) Base Controller: The patterns show more steps to reach the target.

Symphony was easy to analyze and showed higher selection rates close to the center of the traget.

of cursor. Even though it consisted of a gesture component, the path did not show erratic behavior. It was noticeable that most of the path patterns were closer to the distance between the targets. Further, most of the targets were acquired at the center. The path patterns in symphony had ample distance between cursor points in coarse region and little congestion in the fine pointing region. Figure 5.3(c) portrays path patterns for symphony of modalities.

The steps were far more frequent due to small interaction area.

- **Base Controller**: The base controller had a congested path. In many cases the users took a step path opposed to a linear path towards the target. This was caused due to jerks, lifting and repositioning of the finger while interacting on the small surface. The patterns resembled that of the touch interaction, however due to idiosyncrasies in user behavior the step patterns were much frequent than touch. Figure 5.3(d) reveals the cursor behavior using base controller.

5.6.2 Data Analysis

The data collected from the application was statistically evaluated for effects on speed and accuracy of the interaction technique. The main major metrics that were analyzed from the data were the Movement time, Error rate, Overshooting and Throughput. The data was aggregated from the original individual XML files to a table format. Since each block explained in section 5.2 was repeated three times, total number of trials were 3888.

A shapiro-wilcox normality test revealed that data of mean movement time for each interaction technique, ID and block was normally distributed with $p > 0.1$. The next step was to perform statistical tests to verify the effects of the independent variables on the dependent variables.

Mean Movement Time

Movement time was the time taken by the user to move the cursor from the acquired target to the next target and select it. Mean of each trial was considered for analysis.

Movement time MT:

A repeated measures ANOVA with Greenhouse-Geissers corrections revealed significant effect of MODALITY on *Movement Time* (MT) with ($p = 9.276096 \times 10^{-10}$, $F(3,33) = 31.2450174$, $\eta^2 = 0.48488152$).

Modality: significant effect with ($p < 0.05$).

A post-hoc analysis using pairwise t-test with Bonferroni corrections revealed no significant differences between SYMPHONY and TOUCH (TOUCH 7% faster with $mean_{\text{SYMPHONY}} = 3143.342ms$ and $mean_{\text{TOUCH}} = 2909.673ms$) where ($p = 1.0000$). However, there were significant differences between SYMPHONY and GESTURE (SYMPHONY was 77% faster with $mean_{\text{SYMPHONY}} = 3143.342ms$, $mean_{\text{GESTURE}} = 7078.356ms$) where ($p < 2 \times 10^{-16}$), significant differences between SYMPHONY and BASE CONTROLLER (SYMPHONY was 25% faster with $mean_{\text{SYMPHONY}} = 3143.342ms$, $mean_{\text{BASE}} = 4025.121ms$) where ($p = 0.0022$). Further, the test showed significant differences between TOUCH and GESTURE(TOUCH was 83%

SYMPHONY and TOUCH ($p > 0.05$)
SYMPHONY and GESTURE ($p < 0.05$)
SYMPHONY and BASE ($p < 0.05$).

faster with $mean_{\text{TOUCH}} = 2909.673ms$, $mean_{\text{GESTURE}} = 7078.356ms$) where ($p < 2 \times 10^{-16}$), significant differences between TOUCH and BASE CONTROLLER (TOUCH was 32% faster with$mean_{\text{TOUCH}} = 2909.673ms$,$mean_{\text{BASE}} = 4025.121ms$) ($p = 4.2 \times 10^{-5}$) and significant differences between GESTURE and BASE CONTROLLER (BASE 54% faster $mean_{\text{SYMPHONY}} = 3143.342ms$, $mean_{\text{GESTURE}} = 7078.356ms$) where ($p = 2 \times 10^{-16}$).

ID: significant effect with ($p < 0.05$).

A repeated measures ANOVA with Greenhouse-Geissers corrections revealed significant effect of ID on MT with ($p = 8.889821 \times 10^{-6}$, $F(2, 22) = 20.6653455$, $\eta^2 = 0.03198469$).

3.5 and 4.0 ($p < 0.05$), 4.0 and 4.5 ($p < 0.05$), 3.5 and 4.5 ($p < 0.05$) all have significant differences.

A post-hoc analysis using pairwise t-test with Bonferroni corrections showed significant differences between ID 3.5 and 4.0 (3.5 was 8% faster with $mean_{\text{MT:3.5}} = 3920.763$), $mean_{\text{MT:4.0}} = 4263.81$) where ($p = 0.0157$) and significant differences between ID 4.0 and 4.5 (where 4.0 was 9% faster with$mean_{\text{MT:4.0}} = 4263.81$, $mean_{\text{MT:4.5}} = 4682.796$) where ($p = 0.0042$) and revealed significant differences between ID 3.5 and 4.5 (where 3.5 was 18% faster with $mean_{\text{MT:3.5}} = 3920.763$), $mean_{\text{MT:4.5}} = 4682.796$) ($p = 1.3 \times 10^{-6}$).

BLOCK had no effect on MT $p > 0.05$

From this, it could be inferred that there was no learning effect from one block to another. A repeated measures ANOVA with Greenhouse-Geissers corrections revealed no significant effect of BLOCK on MT with ($p = 0.43927$, $F(2, 22) = 0.85416356$, $\eta^2 = 0.00263057$).

A repeated measures ANOVA with Greenhouse-Geissers corrections for interaction between MODALITY $\times$ ID revealed significant effect on MT with ($p = 1.954566 \times 10^{-}5$, $F(6, 66) = 6.51622953$, $\eta^2 = 0.04290363$).

A repeated measures ANOVA with Greenhouse-Geissers corrections for the following interaction revealed no significant effect as shown in Table 5.2.

Interaction	P-Value	F-Statistics	Effect Size (η^2)
MODALITY × BLOCK	0.8852	$F(6, 66) = 0.38640422$	0.00385341
ID × BLOCK	0.692	$F(4, 44) = 0.56173871$	0.00159146
MODALITY × ID × BLOCK	0.4443	$F(12, 132) = 1.00901934$	0.00911671

Table 5.2: Results of in repeated measures ANOVA of different interactions on MT.

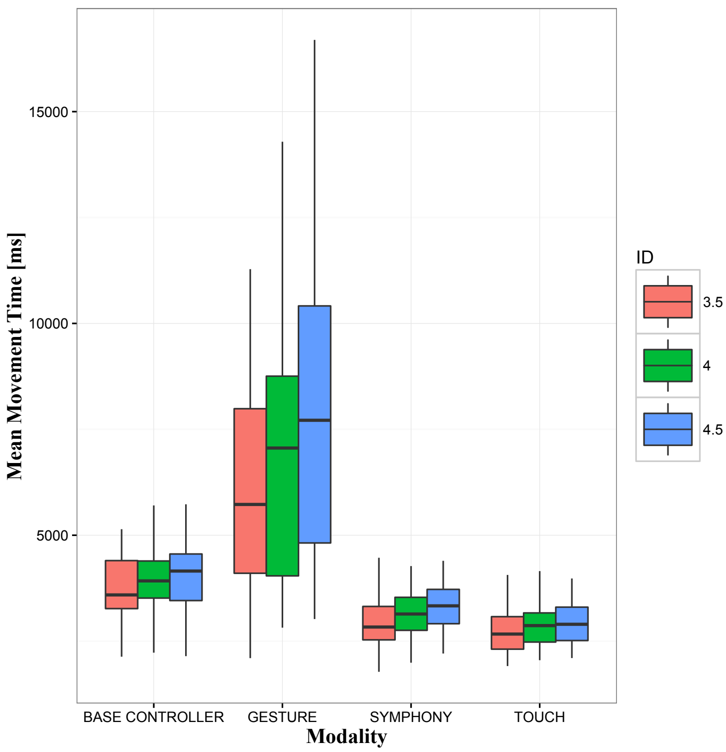


Figure 5.4: Analyzing effect of Technique and ID on Mean Movement Time. SYMPHONY and TOUCH are similar movement time for different ID. These two techniques are lesser compared to other techniques.

TOUCH was faster than SYMPHONY by 7%. However, symphony was substantially faster than gesture by 77% and 25% faster than the base controller.

Through this test, we can understand that symphony of modalities was significantly different compared to other techniques. From Figure 5.4 and comparing the means, we observe that SYMPHONY and TOUCH were the faster among all techniques. Previous experience of users with TOUCH provided an advantage.

From Figure 5.4 we can observe that GESTURE was the most difficult that took longer than all other techniques. Few users were able to perform faster, therefore we can observe higher variance in the data. However, overall movement time on gesture using the smartwatch was much higher than all other techniques. From the test we can also observe that as the ID increased, the movement time also increased.

Mean Error

Definition: *Error*

ERROR:
In this research, error is defined as distance in pixels from the center of cursor to the center of the target at the time of acquisition.

The maximum possible error for largest target was 24 pixels and for the smallest target was 14 pixels. This was the definition because the next target would show only after successful acquisition of the current target.

Modality: significant effect with ($p < 0.05$)

A repeated measures ANOVA with Greenhouse-Geisser corrections revealed significant effect of MODALITY on ERROR ($p = 3.511856 \times 10^{-13}$, $F(3, 33) = 39.6218510$, $\eta^2 = 0.3677129$).

SYMPHONY and TOUCH ($p < 0.05$) SYMPHONY and GESTURE ($p < 0.05$) SYMPHONY and BASE ($p < 0.05$).

A post-hoc analysis using pairwise t-test with Bonferroni corrections revealed significant differences between SYMPHONY and TOUCH (SYMPHONY was accurate by 21% ($MeanError_{\text{SYMPHONY}} = 7.919604px$, $MeanError_{\text{TOUCH}} = 9.787583px$) where ($p = 1.8 \times 10^{-10}$), it also revealed significant differences between SYMPHONY and GESTURE (SYMPHONY was 42% more accurate with $MeanError_{\text{SYMPHONY}} = 7.919604px$, $MeanError_{\text{GESTURE}} = 12.17381px$) where ($p < 2 \times 10^{-16}$), significant differences between SYMPHONY and BASE CONTROLLER (SYMPHONY was 32% more accuate with $MeanError_{\text{SYMPHONY}} = 7.919604px$, $MeanError_{\text{BASE}} = 10.92774px$) where ($p < 0.05$). The test also showed significant differences between TOUCH and GESTURE (TOUCH was 21% more accurate with $MeanError_{\text{TOUCH}} = 9.787583px$, $MeanError_{\text{GESTURE}} = 12.17381px$) where ($p = 5.9 \times 10^{-15}$), significant differences between TOUCH and BASE CONTROLLER (TOUCH being 11% more accurate with $MeanError_{\text{TOUCH}} = 9.787583px$, $MeanError_{\text{BASE}} = 10.92774px$) ($p = 7.8 \times 10^{-5}$) and significant differences between GESTURE and BASE CONTROLLER (BASE being 10% more accurate with $MeanError_{\text{GESTURE}} = 12.17381px$, $MeanError_{\text{BASE}} = 10.92774px$) where ($p = 1.4 \times 10^{-6}$).

A repeated measures ANOVA with Greenhouse-Geisser corrections revealed significant effect of ID on ERROR with ($p = 1.542564 \times 10^{-14}$, $F(2, 22) = 286.979247$, $\eta^2 = 0.51461377$

ID: significant effect with ($p < 0.05$).

A post-hoc analysis using pairwise t-test with Bonferroni corrections showed significant differences between ID 3.5 and 4.0 where ($p = 2 \times 10^{-16}$) (4.0 being 24% more accurate with $MeanError_3.5 = 12.50419px$, $MeanError_4.0 = 9.770472px$), significant differences between ID 4.0 and 4.5 (4.5 is 15% more accurate with $MeanError_4.0 = 9.770472px$, $MeanError_4.5 = 8.331897px$) where ($p = 1.4 \times 10^{-13}$) and significant differences between ID 3.5 and 4.5(4.5 is 40% more accurate with $MeanError_3.5 = 12.50419px$, $MeanError_4.5 = 8.331897px$) ($p < 2 \times 10^{-16}$).

3.5 and 4.0 ($p < 0.05$), 4.0 and 4.5 ($p < 0.05$), 3.5 and 4.5 ($p < 0.05$) all have significant differences.

A repeated measures ANOVA with Greenhouse-Geisser corrections revealed no significant effect of BLOCK on ERROR $p = 0.52037$ and $F(2, 22) = 0.3590721$ $\eta^2 = 0.0019891546$.

BLOCK had no effect on ERROR as $p > 0.05$.

A repeated measures ANOVA with Greenhouse-Geisser corrections for interactions of MODALITY and ID had significant effect on ERROR with ($p = 2.294069 \times 10^{-7}$, $F(6, 66) = 9.947448$, $\eta^2 = 0.14707128$).

A repeated measures ANOVA with Greenhouse-Geisser corrections for interactions that have no significant effect on ERROR are summarized in Table 5.3.

Interaction	P-Value	F-Statistics	Effect Size (η^2)
MODALITY × BLOCK	0.7011	$F(6, 66) =$ 0.6735290	0.0087042
ID × BLOCK	0.9141	$F(4, 44) =$ 0.56173871	0.24547753
MODALITY × ID × BLOCK	0.7713	$F(12, 132) =$ 0.96122234	0.02058343

Table 5.3: Results of in repeated measures ANOVA of different interactions on ERROR.

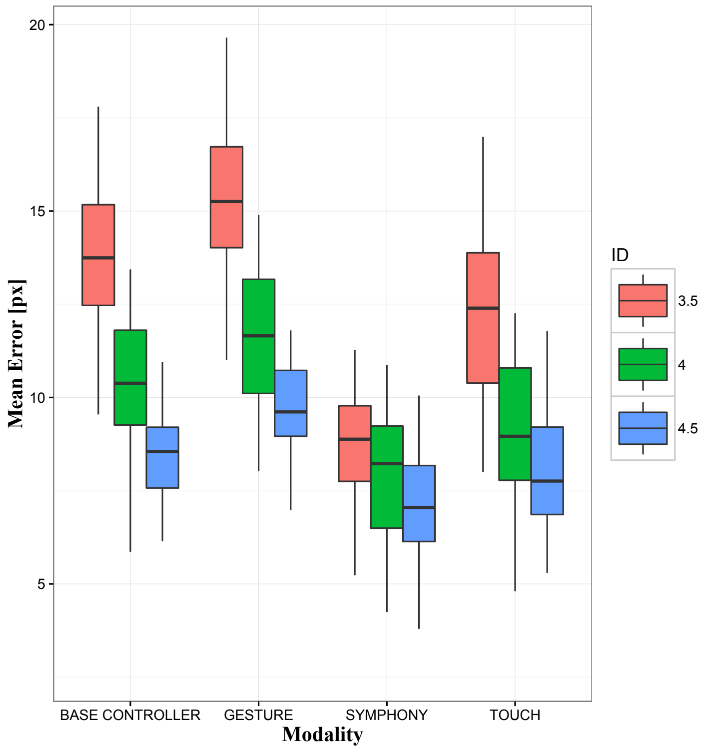


Figure 5.5: Effect of MODALITY and ID on MEAN ERROR. The variance and mean are smaller for smaller targets.

The accuracy from means, SYMPHONY $> 21\%$ than TOUCH, $> 42\%$ than GESTURE and $> 32\%$ than BASE.

From Figure 5.5, SYMPHONY has the least error rate in all target acquisition tasks. Accuracy is inverse of error. As observed, SYMPHONY is the most accurate of the techniques.

Overshoot

OVERSHOOT:
in this research, overshoot is a logical value that determines whether the user had previously moved the cursor into the acquisition area of the target and failed to acquire it in the first try but acquired the target later.

Definition: *Overshoot*

The overshoot could only be accounted if the user had glanced over and away from the target with the cursor. If the user missed the target in the first attempt, Wobbrock et al. [2008] suggest that the fine control of the interaction technique has been affected. OVERSHOOT was a dichotomous variable.

Overshoot determines better or worse fine control for the interaction technique.

A Cochran's Q test was conducted. Cochran's Q test found that there exists significant effect on OVERSHOOT in among the different interaction techniques tested ($\chi^2(3) = 246.71$, $p < 0.05$).

A post-hoc pairwise comparison using continuity-corrected McNemar's tests with Bonferroni correction revealed significant less number of overshoots in SYMPHONY as compared to GESTURE where ($p < 0.05, \phi = 0.060797$). However, there were no significant effect between SYMPHONY and TOUCH where ($p > 0.05, \phi = 0.0001281$), and no significant effect between SYMPHONY and BASE CONTROLLER where ($p > 0.05, \phi = 0.0003588$).

Overshoot count TOUCH: 160, GESTURE: 397, SYMPHONY: 169 and BASE: 154.

The observation of this test reveals that GESTURE lacked fine control. The users were not able to acquire the target even though they were on the target. Further, due to the total travel distance higher in GESTURE, the lack of fine control could be established. Figure 5.6 and the overshoot count reveal significantly higher overshoots in GESTURE technique compared to other techniques confirming the low fine control.

BASE CONTROLLER, TOUCH and SYMPHONY had equally good fine control.

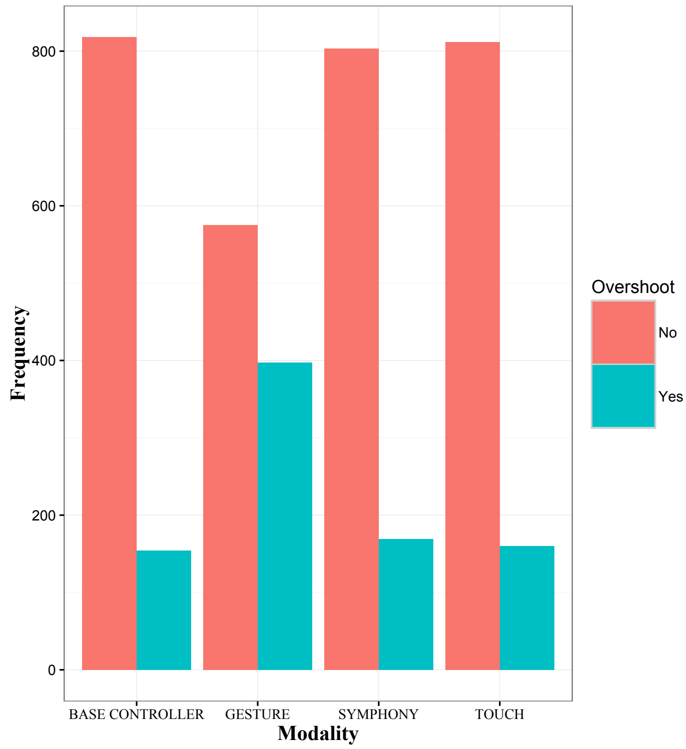


Figure 5.6: Plot of overshoot influenced by interaction technique. SYMPHONY, TOUCH and BASE CONTROLLER do not have significant differences in overshoot

Throughput

Throughput abstains from influence of difficulty.

Throughput of the interaction technique is defined as the ratio of the effective index of difficulty ID_e to the movement time.

$$Throughput(TP) = \frac{ID_e}{MT}$$

Modality: significant effect with $(p < 0.05)$

A repeated measures ANOVA with Greenhouse-Geisser corrections revealed significant effect of MODALITY on THROUGHPUT ($p = 5.324669 \times 10^{-13}$ and $F(3, 33) = 55.6324373$, $\eta^2 = 0.4848685$).

A post-hoc analysis using pairwise t-test with Bonferroni corrections revealed no significant differences between SYMPHONY and TOUCH($MeanThroughput_{\text{SYMHONY}} =$

$1.877823 \times 10^-3 bits/ms$, $MeanThroughput_{\text{TOUCH}} = 1.931579 \times 10^-3 bits/ms$) where ($p = 0.58$), however, showed significant differences between SYMPHONY and GESTURE ($MeanThroughput_{\text{SYMHONY}} = 1.877823 \times 10^-3 bits/ms$, $MeanThroughput_{\text{GESTURE}} = 1.019498 \times 10^-3 bits/ms$) where ($p < 2 \times 10^{-16}$) and significant differences between SYMPHONY and BASE CONTROLLER ($MeanThroughput_{\text{SYMHONY}} = 1.877823 \times 10^-3 bits/ms$, $MeanThroughput_{\text{BASE}} = 1.429942 \times 10^-3 bits/ms$) where ($p < 2 \times 10^{-16}$). Further, the test also showed significant differences between TOUCH and GESTURE ($MeanThroughput_{\text{TOUCH}} = 1.931579 \times 10^-3 bits/ms$, $MeanThroughput_{\text{GESTURE}} = 1.019498 \times 10^-3 bits/ms$) ($p < 2 \times 10^{-16}$), significant differences between TOUCH and BASE CONTROLLER ($MeanThroughput_{\text{TOUCH}} = 1.931579 \times 10^-3 bits/ms$, $MeanThroughput_{\text{BASE}} = 1.429942 \times 10^-3 bits/ms$) where ($p < 2 \times 10^{-16}$) and significant differences between GESTURE and BASE CONTROLLER ($MeanThroughput_{\text{GESTURE}} = 1.019498 \times 10^-3 bits/ms$, $MeanThroughput_{\text{BASE}} = 1.429942 \times 10^-3 bits/ms$), where ($p < 2 \times 10^{-16}$).

SYMPHONY and TOUCH no effect with ($p > 0.05$) SYMPHONY and GESTURE ($p < 0.05$) SYMPHONY and BASE ($p < 0.05$).

A repeated measures ANOVA with Greenhouse-Geisser corrections revealed significant effect of BLOCK on THROUGHPUT ($p = 5.830675 \times 10^{-3}$ and $F(2, 22) = 6.5593792$, $\eta^2 = 0.0097912$).

Block: significant effect with ($p < 0.05$)

A pairwise t-test revealed no significant effect between BLOCK 1 and 2 ($MeanThroughput_{b1} = 1.601403 \times 10^-3 bits/ms$, $MeanThroughput_{b2} = 1.580265 \times 10^-3 bits/ms$) where ($p = 0.9294$), however, showed significant effect between BLOCK 2 and 3 ($MeanThroughput_{b2} = 1.580265 \times 10^-3 bits/ms$, $MeanThroughput_{b3} = 1.512463^10 - 3bits/ms$) where ($p = 0.00887$)and showed significant effect between BLOCK 1 and 3 ($MeanThroughput_{b1} = 1.601403 \times 10^-3 bits/ms$, $MeanThroughput_{b3} = 1.512463^10 - 3bits/ms$) where ($p = 0.00036$).

BLOCK1 and 2 not significant with ($p > 0.05$), 2 and 3 ($p < 0.05$), 1 and 3 ($p < 0.05$) have significant differences.

A repeated measures ANOVA with Greenhouse-Geisser corrections revealed no significant effect of ID on THROUGHPUT ($p = 0.22973$, $F(2, 22) = 1.5737109$, $\eta^2 = 0.0023816$).

ID has no significant effect on THROUGHPUT with ($p > 0.05$).

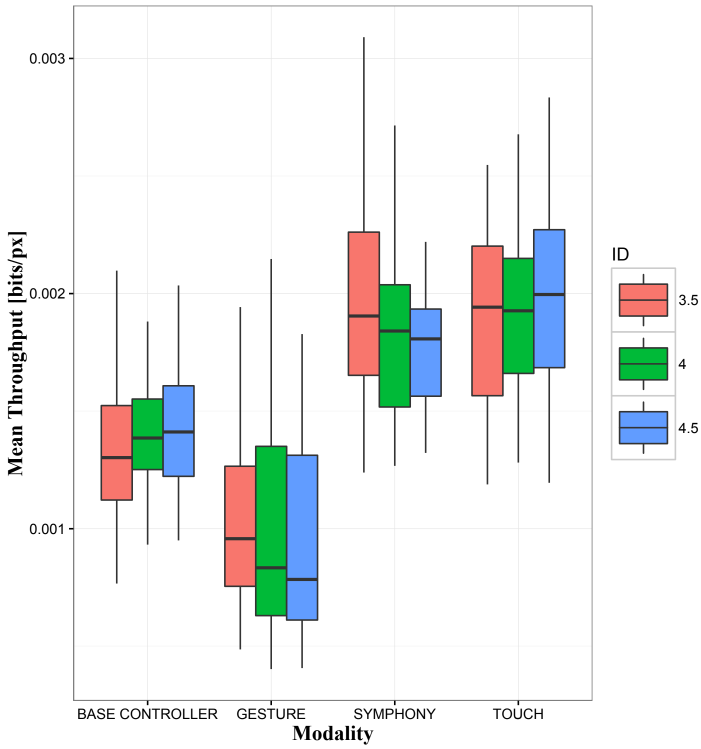


Figure 5.7: Analyzing effect of TECHNIQUE and ID on THROUGHPUT of the interaction technique. Touch has the highest throughput followed by symphony and base controller

A repeated measures ANOVA with Greenhouse-Geisser corrections for interaction between MODALITY $\times$ ID showed significant effect on THROUGHPUT where ($p = 1.506231 \times 10^{-5}$, $F(6, 66) = 6.6694200$, $\eta^2 = 0.0240968$).

A repeated measures ANOVA with Greenhouse-Geisser corrections for interaction between MODALITY $\times$ BLOCK showed significant effect on THROUGHPUT where ($p = 0.00177$, $F(6, 66) = 2.7886589$, $\eta^2 = 0.0110420$). A repeated measures ANOVA with Greenhouse-Geisser corrections did not show any effect on THROUGHPUT from the interactions summarized in Table 5.4.

SYMPHONY has highest throughput for larger targets.

Figure 5.7 shows that SYMPHONY and TOUCH have the highest throughput compared to other techniques even in

Interaction	P-Value	F-Statistics	Effect Size (η^2)
ID × BLOCK	0.60532	$F(4, 44) = 0.6862315$	0.0012781
MODALITY × ID × BLOCK	0.33408	$F(12, 132) = 1.1394736$	0.0059124

Table 5.4: Results of in repeated measures ANOVA of different interactions on THROUGHPUT.

higher values of ID. Therefore, as from the Figure 5.7,. The least throughput was with GESTURE, making it the least popular interaction technique.

5.6.3 Retrospection

The videos were investigated to reveal user behavior with each technique. The videos revealed critical influences of technique such as user behavior, expression and comments during the tasks.

Proxemics refers to the spatial requirement for humans.

Touch

The users showed a similar behavior in TOUCH technique where the dominant hand was used to interact with the smartwatch except for U6. All the participants used the forefinger to interact with the surface with their arms bent inwards towards the body as shown in Figure 5.8(a). This was due to the previous knowledge they obtained from smartphone usage. There were idiosyncratic behaviors observed where the users flicked their finger rather than persistent control of the cursor. This could have been an effect of no control display ratio adaptation in touch. The proxemics of interaction for touch was lower as the users nudged their arms inwards towards their body.

Users subconsciously used the fore finger to interact.

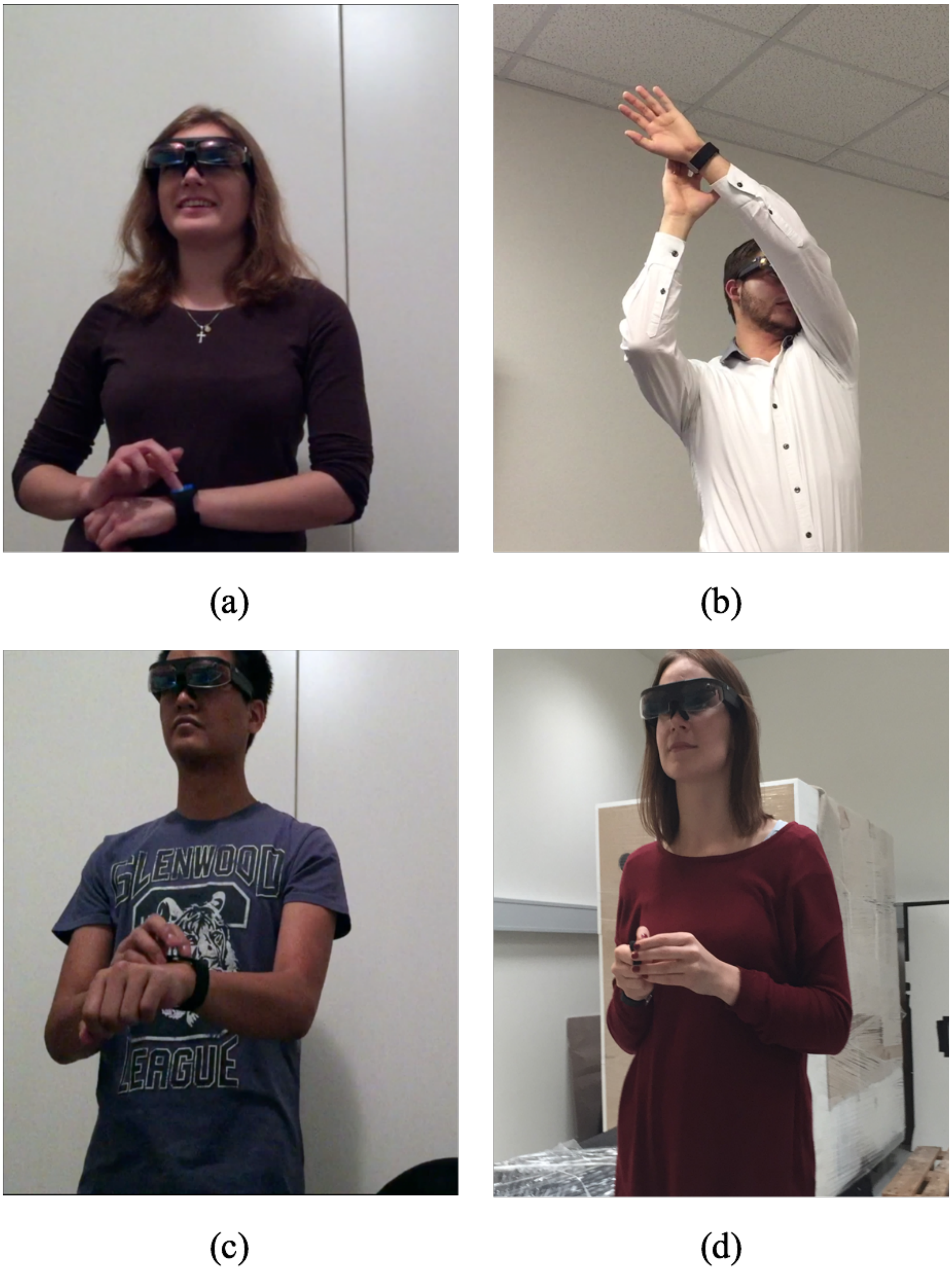


Figure 5.8: Snapshots of videos from retrospection. (a) **Touch**: Shows idiosyncratic behavior of forefinger to interact with the smartwatch. (b) **Gesture**: User's arm is intruding the FoV of the user during interaction. (c) **Symphony**: Proxemics of interaction and movement of arms is lesser than GESTURE and both hands actively involved to reduce costs for switching between techniques. (d) **Base controller**: Shows user using in bi-manual mode instead of afforded uni-manual. The user is stabilizing the movement by using both hands.

Gesture

The GESTURE technique was more obtrusive and users constantly questioned if it made them uncomfortable to use in front of an investigator. Few users adhered to the prescribed positioning of the arm, whereas others subconsciously changed their position by moving their arms inward. This caused the technique to behave unexpectedly as the change in orientation of the smartwatch caused changes in axes for gesture recognition. The proxemics of interaction was very high for users except U3, U9 and U12.

Most participants spoke out, “I must be looking funny doing this”.

The users found it difficult to perform this task. U8 expressed *"I don't like to move my arm."* and controlled the cursor using wrist movements, that took longer than the standard GESTURE. U9 mentioned *"I think moving the upper body makes the task easier."* and moved her torso in coordination with her arm. Since the arm movement reached above the shoulder level for few targets, the arm intruded in the FoV of the user as shown in the Figure 5.8(b). Clutching was used by all the users when the maximum degree of arm movement had reached. U7 used clutching to position the arm to the perpendicular posture and bring the cursor to the center before beginning the task. U9 used upper torso movement in coordination with the arm movement to benefit easier and farther traversal of the cursor.

Symohony

Evaluating the videos of SYMPHONY revealed different user behaviors. The users substantially benefited from the faster movement of the cursor. All users performed the tasks in a standard touch axes adaptation technique. When their arm position was completely inwards the touch orientation changed. All users except U8 and U11 followed the adaptive touch. However, since these users adhered to the prescribed position, the axes adaptation was not required and there is no visible evidence of this in the videos. Participants felt comfortable with this technique. U6 expressed *"I like this and I think I can get used to it using everyday."* On other hand, U8 preferred interaction with just one arm. U2

The prescribed position of arm is perpendicular to the body.

was expressed his surprise by *"I can't believe how fast I finished the task."*

There was no need for adaptive axis inversion.

U8 an U11 refused to use the inverted touch since they claimed that the cursor behavior was unexpected. This was due to the arm folded inwards by users as opposed to the prescribed position. Most users mastered the technique within a single test round. Further, the proxemics of interaction was not high as in GESTURE technique. All users always accompanied the hand for touch with the smartwatch. This was to limit the switching costs between the techniques as shown in Figure 5.8(c). The maximum upward movement of the arm was till the shoulder.

Base Controller

Was not comfortable for users with larger fingers.

The BASE CONTROLLER or called the ring controller because it could be worn as a ring to the finger. This technique showed few anomalies in the usage pattern. As this device was supplied by the manufacturer, the acceleration of cursor provided by the operating system was turned off. This felt cumbersome to the users as they made multiple swipes on the interaction surface to move the cursor. In addition, the interaction area was really small. Although the device was designed for uni-manual interaction, few users preferred wearing the device on a finger of non dominant hand and interacting with a finger of the dominant hand as seen in Figure 5.8(d).

5.6.4 Questionnaire and Interview

Following the user study the participants were requested to fill the questionnaire consisting of questions that would reveal their preference of interaction technique. The two most important values of interest were confidence scores and difficulty score.

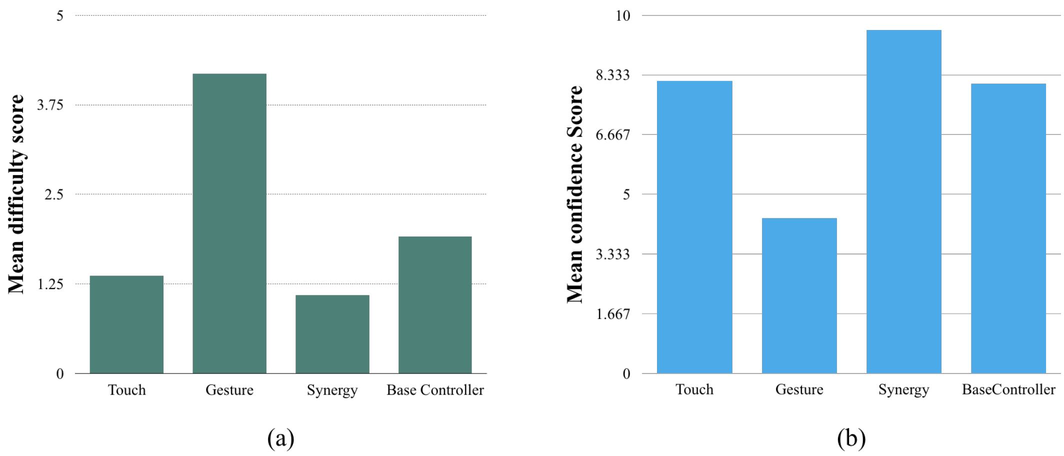


Figure 5.9: Plots derived from the user questionnaire. (a) Shows the mean confidence scores for each interaction technique. (b) Shows the mean difficulty score for each interaction technique.

The confidence score was a rating between 0 and 10 denoting how confident were the users while acquiring the target at its center. The other dimension of the confidence score was to understand if the cursor behaved according to the user. The value 0 shows no confidence and value of 10 shows completely confident. Figure 5.9(a) reveals that the participants felt most confident in SYMPHONY having an rating ($m_c = 9.583$). GESTURE technique was the most unpopular with a confidence score of ($m_c = 4.33$) compared with all the interaction techniques amongst all users.

SYMPHONY had the best average confidence scores compared to all techniques.

Difficulty score was also a value rated between 0 to 5 by the users for an interaction technique. This value was influenced by the difficulty to understand and adapt to the interaction technique, difficulty to control the cursor in the technique. As earlier, 0 denotes no difficulty and 5 denotes most difficult. Figure 5.9(b) shows users felt TOUCH and SYMPHONY techniques were least difficult with a score of ($m_d = 1.091$) compared to other techniques. The most difficult was the GESTURE technique with a score ($m_d = 4.182$) engineered using the smartwatch.

GESTURE using smartwatch was the most unpopular technique.

Post user study, the participants were asked routine questions based on the experiment to understand their experience with each of the interaction technique. Most users

Few comments are excerpt of answers to the questions.

there too favored the idea of SYMPHONY. The inverted touch mapping bothered few users. U1 stated *"The touch movements was a little iffy due to the position of the hand."*. U2 was uncomfortable with fine pointing and expressed *"I recommend removing the "slow down" feature to make the technique robust."*. (See Appendix B "User Study and Analysis" for all the comments).

The user preference was SYMPHONY followed by TOUCH followed by BASE CONTROLLER and at last GESTURE.

However, most users agreed that upon more number of practices they could master SYMPHONY of modalities and feel more comfortable while using it. Since all users had prior experience using touch devices, the touch technique was also popular amongst users. No user reported to have problems with TOUCH technique. On the contrary, most users disliked GESTURE as it involved lot of arm/body movement. However, users familiar with gesture technique were able to perform with ease except U1. The BASE CONTROLLER was popular with two users over SYMPHONY and TOUCH. They claimed that the ability to use the device with just one hand while other hand remains free is quite useful than indulging both hands to interact.

5.7 Summary

The interaction technique SYMPHONY of modalities was evaluated using three different methods. The statistical method revealed that SYMPHONY was best suited for faster and accurate acquisition while TOUCH was best suited for fast target acquisition. The GESTURE technique was very unpopular and suffered due constraints that had to be adhered for ease of interaction. The path patterns of SYMPHONY was also clean and closer to the distance between targets. Upon retrospection the proxemics of interaction did not pass above the shoulder level for the SYMPHONY while GESTURE had higher proxemics. Finally, a questionnaire revealed that most users favored SYMPHONY of modalities over all other techniques.

Chapter 6

Summary and Future Work

"We can only see a short distance ahead, but we can see plenty there that needs to be done."

—Alan Turing

This thesis is an attempt to provide an out of the box input solution for HWDs using collocated devices. The thesis successfully evaluates the benefits of repurposing a smartwatch as an input controller for HWDs. This device could be used as input controller for small and large FoV HWDs alike. Benefit of using the smartwatch is that it is a wearable. In addition, the smartwatch is a masquerading technology that is powerful for both designers and engineers.

RQ: How to design a smartwatch as an input controller.

6.1 Summary and contributions

This thesis began with a broader research perspective of developing an interaction technique for HWDs. In order to find the best suited input controller a survey of different possible input controllers were done. The smartwatch was a eventual choice for the input controller. Over the period of this thesis, smartwatch was repurposed as an input controller for a smaller FoV consumer smartglass. This involved navigating menu based interface. This research extended to cursor based technique for larger FoV. Cursor control using the smartwatch was achieved by designing a novel interaction technique. The smartwatch was designed to recognize both touch and gesture technique and symphony of touch and gesture.

Preliminary interaction design used indirect touch.

The initial interaction design involved the touch area of the smartwatch being designed to recognize input and control a menu based interface on a smartglass. The technique was designed after studying, which different watch faces yields better proprioception. Further, awareness about the boundary conditions was also studied as the watch leaves a kinesthetic imprint on the wrist of the watch wearing person. With help of the data form the user study, usage patterns of touch-gestures depending on the affordance of the watch face were summarized.

Due to system constraints smartphone was used as a data-relay.

The software designed to run the smartwatch as an indirect touch input controller used a smartphone for wireless relay of events from smartwatch to the smartglass. This was accomplished by running smartglass as a UDP server to which the smartphone connected and relayed the events. This technique was a bi-manual interaction where one hand was left unused. This led us to the observed *still hand problem* through which the foundations for a novel interaction technique was laid.

Larger FoV HWDs commonly use cursor input as an efficient method to interact with the operating system. The symphony of modalities was designed to exploit cursor technique for such larger displays.The technique combined best characteristics of touch and in-air gesture techniques.

Symphony of modalities is a novel interaction technique.

The critical part of developing symphony was to provide faster and efficient cursor movement. In doing so, we could tackle the existing problems of the interaction techniques such as small interaction space, having to carry an additional hardware, being highly obtrusive or slower and imprecise cursor movement. Therefore, this technique was developed to (1) solve the *Still hand problem* by indulging both hands, (2) increase the interaction space by using gesture, (3) Making it extensible for larger display sizes, (5) provide state switching without additional costs (4) retaining the small portable device i.e. the smartwatch for interaction.

Designing the smartwatch to recognize touch followed a touchpad metaphor. Cursor control was established by interaction on the screen of the smartwatch. Gesture on the smartwatch required more engineering as sensors were prone to errors and drifts. With help of sensor fusion, the gesture control was developed. The combination of these two techniques was asynchronous, where gesture technique was used for *coarse* pointing and touch technique for *fine* pointing.

Fast gesture for coarse pointing and slow touch for fine pointing.

This technique leveraged using gesture to move closer to the target with lesser time and with touch, precisely acquire the target. In symphony, the switching costs between coarse and fine pointing were minimized by dynamically adapting the pointer speed as it neared the target. Slowing the pointer down near the target, was a visual feedback to instigate to switch to the fine pointing state. Feedback from the ST-HWD was a discrete vibration upon successful target acquisition.

The participants took part in a user study and were asked to perform a target acquisition task with each technique while corresponding technique's data was recorded. The session were video recorded to observe user behavior with each of the technique. The differences in behavior from one technique to another was reviewed and discussed.

Symphony proved to be better than its counterparts.

Symphony was tested with its individual counterparts. Gesture was the most difficult and took more time compared to all techniques. The Base Controller's pointer movement was sticky, thus, the time and accuracy suffered. The final results revealed that touch and symphony were equally faster in the target acquisition tasks. However, symphony was better in terms of accuracy. The users also preferred symphony of modalities for interacting with the HWDs. In addition to the user's preference, the average confidence scores were highest for symphony followed by touch.

6.2 Future work

This thesis was the first step to explore the validity of the symphony of modalities. The focus was on how an interaction technique could be developed and later evolve into a better technique by repuroposing existing technology. The applications for using the smartwatch as an input controller are many.

A preliminary system showcased the use of smartwatch as a purely touch controller to navigate a menu on the smartglass. This could be extended to study how using bi-manual interactions could influence menu traversal in larger FoV HWDs.

Altering the parameters of hardware.

The main part of the thesis research uses a set of parameters that were predefined to evaluate the technique. The influence of parameters such as the control display ratio was not studied in detail. During the thesis we set values for control display ratio to be greater than 1 for coarse pointing and less than 0.5 for fine pointing. The research can be furthered by varying the control display ratio and verifying the validity of this technique for different values of control display ratio.

Further evaluating effect of proprioception.

The technique used a prescribed in-air gesture model for coarse interaction. The technique could be extended to study various postures of the arms that affect the cursor control. Since the user has an immersive experience, she has to rely on proprioception. Therefore, extending the research to analyze importance of proprioception in such interaction techniques for HWDs is vital.

Using different gesture techniques.

The in-air gesture technique evaluated was designed using sensor fusion on the smartwatch, proved to have many drawbacks. Further investigation could be done by integrating other effective in-air gesture techniques in symphony together with different gesture sets. Instead of whole arm movement, only wrist movement or finger movement could be combined with touch to study a new symphony of modalities. The cognitive effort required to learn and use such a technique should also be evaluated.

Exploring symphony against wall size displays.

The thesis focused on large FoV device stretching across 30° or 65 inches diagonal display. Further exploration of using this technique in large wall sized displays as in Nancel et al. [2013] could reveal interesting details in user behavior. The limited FoV of the human ocular system to interact with information on large displays using upper torso movement along with symphony could be a potential setup. However, the cognitive effort required to perform such a task in coordination would have to be studied.

Real world applications in AR need to be developed.

Our work showed combination of touch and in-air gesture. The study focused on the effects of only two techniques. Further investigation into effects of introducing another interaction technique is important to analyze the dynamics between touch and gesture. It would also determine whether this technique would be valid if there was another interaction technique in this synergy. Future efforts can include how to use multiple techniques in synergy to yield a novel interaction technique.

The applications of such an interaction technique for ST-HWD are plenty. In production, this technique could be used to control robotic arms remotely while visualizing the effects on the HWD. It could further be used for hierarchical menu navigation. The gesture could be mapped for faster traversal of the parent menu and touch could be used for entering the hierarchy and selecting the desired menu item.

Potential applications in gaming using HWDs.

In gaming, the HWD and smartwatch compliment each other. In order to provide the user a clean immersive gaming experience with HWD where the smartwatch could track user position in 3D and symphony of modalties provides coarse and fine targeting. The accuracy provided by this technique could be used to design interesting user experiences. Finally, this technique provides an open canvas for designers to exploit various types of interaction mappings and user experiences.

Appendix A

Application Development

A.1 Fitts Law

The standard Fitts Law was proposed by Paul Fitts in 1956 that determines total movement time required for a target acquisition using a metric to define difficulty in bits. This was called the Index of Difficulty (ID). The formula was given as:

Fitts law is ISO 9241 standard.

$$ID = log_2(2\frac{D}{W})$$
$$MT = a + b \times ID \tag{A.1}$$

Where *MT* is the total movement time and *a* and *b* are empirically determined constants. Fitts Law Task following the Shannon's approximation of Fitts Law proposed by McKenzie using the Shannon-Harley's theorem MacKenzie [1992]. Shannon's approximation for Equation A.1 which is given by:

$$MT = a + b \times log_2(\frac{D}{W} + 1) \tag{A.2}$$

This equation is a ISO 9241 standard that is used in as the foundation for empirical research in this thesis.

Throughput

Throughput (TP) is a composite measure of both the speed and accuracy of performance. The measure is calculated in bits/ms. As opposed to movement time or error rate, throughput is relatively independent of the task difficulty.

Throughput is the ratio of the effective index of difficulty derived from effective width W_e to movement time. Effective width that captures spatial variability is calculated as

$$W_e = 4.133 \times \sigma_x$$

σ_x is the standard deviation in selection coordinates measured along the axis of approach or in our case the ERROR. The ID_e is obtained through W_e. Therefore, the throughput is calculated as:

$$ID_e = log_2(\frac{D}{W_e} + 1)$$

$$TP = \frac{ID_e}{MT} \tag{A.3}$$

A.2 Limitations

The inherent limitations of the system for the **initial prototype** is described here. These are the various limitations that led to decisions in the design process.

Hardware Limitations

Device restrictions were very apparent.

The system on a chip (SoC) drained substantially drained the battery running simple applications and heated up quite fast.The battery sustainability or heat from the device influences the overall user experience and thus, the software needs to be optimized.

The smartwatch used was a Sony Smartwatch 3 running AndroidWear operating system. Smartwatches are built as complementary devices for smartphones. A smartwatch must always remain paired to the phone to exchange data. The Bluetooth stack of the smartwatch was not directly accessible and WiFi connection was derived from the phone's connectivity.

Software Limitations

The operating system on the smartglass prototype was a basic linux operating system that had access to only a few system services. Communication between devices needed to be established for interaction technique to respond. The Bluetooth stack on the smartglass was limited to one predefined device. In this case, every prototype required a unique smartwatch. Further, if one smartwatch fails, the corresponding prototypes' software had to be rebuilt.

The applications available on system are a web browser and a video player. The web browser handled basic webpages. It did not support on HTML 5 and CSS 3D required for rich user interfaces. The video player was capable of playing only x264 encoded videos compacted to a certain resolution. The smartwatch ran the AndroidWear operating system which is based on AndroidWear is based on the Android's kernel by limiting system services. The Bluetooth mechanism here followed a different architecture. A shared data layer between the smartphone and paired smartwatch was key to data transfer. The Bluetooth stack was designed in such a way that communication was encapsulated by the common data layer.

Bluetooth Stack

The bluetooth stack of the AndroidWear was designed for communication only with paired smartphone. The Bluetooth stack is as shown in Figure A.1. As the smartwatch generated the data, it was pushed into the data layer of the smartwatch. An application on the smartphone that realizes the services by the paired smartwatch introduces the

The Bluetooth communication from the smartwatch was possible only via data layer.

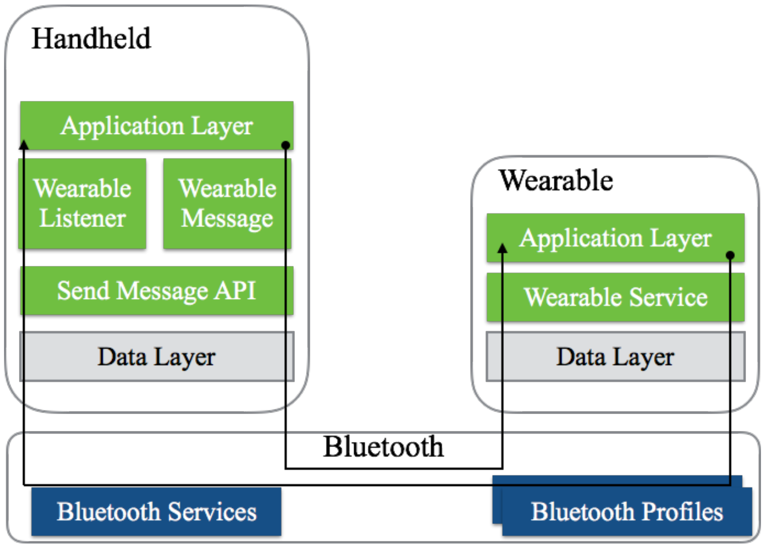


Figure A.1: Bluetooth architecture and communication between smartwatch and smartphone via the data layer.

data layer. The data is communicated via Bluetooth channel.

TCP vs UDP

TCP vs UDP, where UDP was chosen over TCP

In TCP, if the packets were not sent successfully, the client would retransmit. This causes a delay given that two devices were communicating over WiFI. As observed TCP the packets would trigger the scroll seconds after the user had finished scrolling. The interaction felt post-realtime. Thus, the reason for choosing UDP as the protocol of choice was vital for user experience. In UDP, if sending of packets was unsuccessful, the packets was dropped instantly instead of retransmitting. Therefore, there was no overhead of retransmission in UDP. The user input felt instant and thus drastically improved user experience.

Gesture Recognition

The gesture recognition was not possible to implement just with individual sensor data such as the accelerometer. Determining the user's arm position based on acceleration data in the direction of gravity is done by subtracting the gravity vector component from the data that yields the acceleration in a particular direction. Standard physics suggests by integration acceleration, velocity is obtained and upon another integration position in 3D is obtained. The formula is summarized in Equation A.4

$$\vec{p} = \int \int \vec{a} \tag{A.4}$$

where $\vec{p}$ is the position vector and $\vec{a}$ is the acceleration vector after subtracting gravity. Drawback of this method is that the accelerometer sensor readings are not precise that leads to larger errors in position values. Using a low pass filters or Kalman filters, the noise in data caused due to other sensors is reduced but it is not effective against errors induced by integration drift. To overcome this problem, we make use of sensor fusion.

A.3 Software Engineering

The software engineering strategies apart from the architecture are discussed in this section. The two software applications running on the ST-HWD and the application running on the smartwatch were developed to be optimized.

Platform

The software development for the final interaction design was completely done on Android Java. The smartwatch was powered by AndroidWear, a slimmer version of Android and the ODG R6 ST-HWD was powered by customized version of Android for see-through display called the *Reticle OS*.

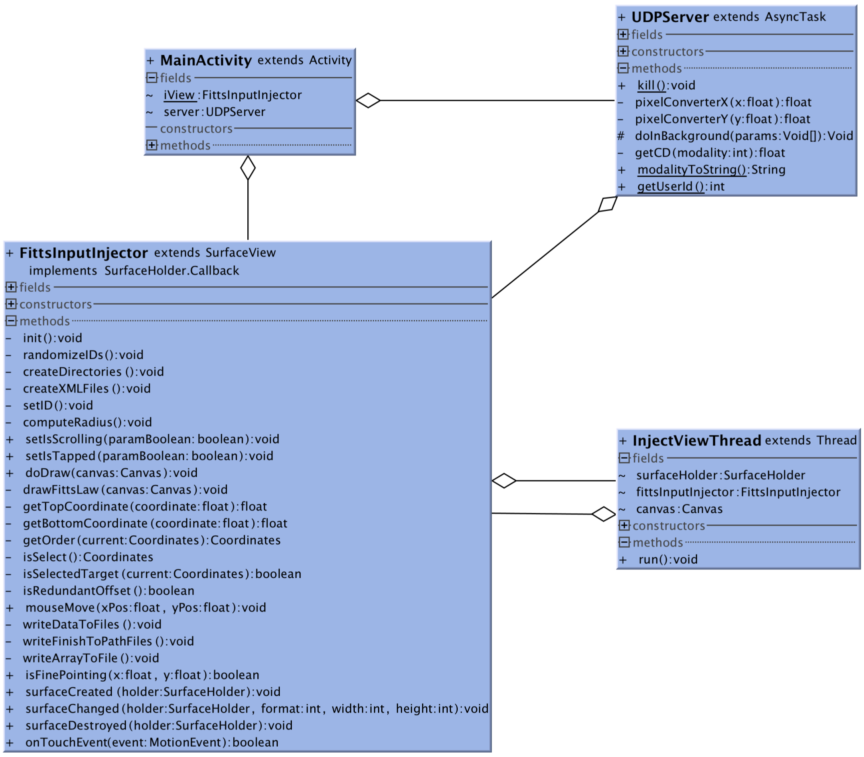


Figure A.2: Shows class diagram for the Fitts experiment.

Fitts Layout

The software developed for the Fitts layout was an Android application that was designed for Android 4.2. Figure A.2 shows the design of the application with help of the class diagram.

Software class design for Fitts application

The FittsInputInjector is the a SurfaceView class and InjectViewThread class is a background renderer that runs as an asynchronous task. This manager is in charge of rendering the FittsInputInjector class that will handle drawing the layout, cursor and the behavior of the cursor.

UDP server class is a background task that will constantly listen to packets that are being sent. Upon receiving the packets, the server will decode the packets and map the control display to each of the techniques. If the system responds purely to touch input the control display ratio multiplier is 0 for gesture recognizer. In symphony, the control

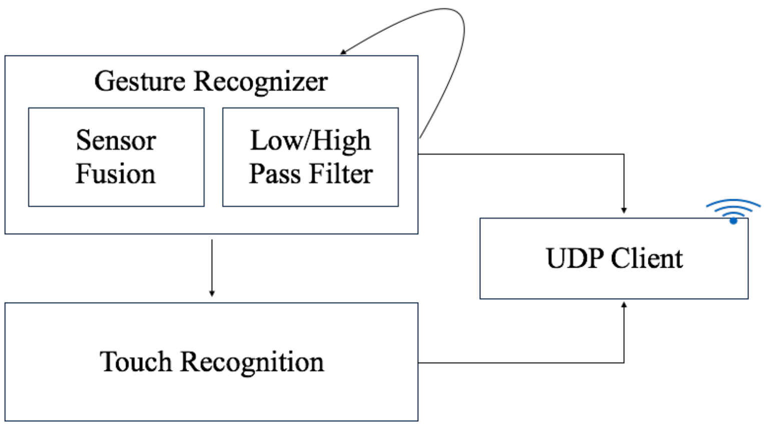


Figure A.3: State diagram for both gesture and touch recognition using smartwatch. Gesture is toggled off in touch state until then gesture recognition keeps running in the background.

display ratio for gesture is predefined to be greater than 1 and adapting as the cursor moves away from or closer to the target. The MainActivity is a holder for the SurfaceView.

Smartwatch Software Design

The Smartwatch AndroidWear application was capable of reading touch and gesture input. The gesture recognizer was a separate thread to optimize the software for speed and battery efficiency. Here, the touch recognition was active only on when the finger was placed. On the other hand the gesture recognizer was always active. Figure A.3 shows the state diagram of the software design used for the application that was designed to recognize both touch and gesture.

AndroidWear 5.1 version was used to develop the system.

The gesture recognizer was toggled off/on when the finger was placed or removed from the touch screen of the smartwatch. This was achieved by using a thread lock and release method with Java's synchronize capability.

A.4 Algorithms

The algorithms for recognizing touch followed the typical mouse movement algorithm. The difference between previous and current position is used to determine the direction of the cursor. The Android code snippet for touch recognition is:

Code snippet for touch recognizer:

```
public boolean onTouch(MotionEvent event)
{
   switch(event)
   {
        case MotionEvent.ACTION_DOWN:
           // initialize the previous x
           // and y points of press
           float initX = event.getX();
           float initY = event.getY();
           return true; // Event has been consumed

        case MotionEvent.ACTION_MOVE:
           // subtract previous x and y and
           // current x and y position
           float curX = initX - event.getX();
           float curY = initY - event.getY();
           sendEventToGlass(curX, curY);
           return true; // Event has been consumed
    }
}
```

Sensor Fusion

The most vital implementation for gesture recognition using smartwatch was the sensor fusion. The common way to get the altitude of an Android device is to use the SensorManager.getOrientation() method to get the three orientation angles. These two angles are based on the accelerometer and magenotmeter output. In simple terms, the acceletometer provides the gravity vector (the vector pointing towards the center of the earth) and the magnetometer works as a compass.

The Information from both sensors is sufficient to calculate the device's orientation. However both sensor outputs are inaccurate. The gyroscope in the device is far more accurate and has a very short response time. Its downside is the dreaded gyro drift. The gyro provides the angular rotation speeds for all three axes. To get the actual orientation those speed values need to be integrated over time. This is done by multiplying the angular speeds with the time interval between the last and the current sensor output. This yields a rotation increment. The sum of all rotation increments yields the absolute orientation of the device. During this process small errors are introduced in each iteration. These small errors add up over time resulting in a constant slow rotation of the calculated orientation, the gyro drift.

Magnetic field sensor output includes a lot of noise.

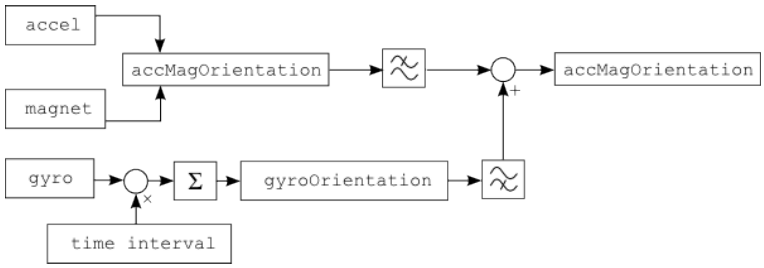


Figure A.4: Shows the mechanism of sensor fusion. High pass filter for gyroscope data and low pass filter for accelerometer and magnetometer data.

Drifts in calculation is caused by gyroscopic white noise.

To avoid both, gyro drift and noisy orientation, the gyroscope output is applied only for orientation changes in short time intervals, while the magnetometer/acceletometer data is used as support information over long periods of time. This is equivalent to low-pass filtering of the accelerometer and magnetic field sensor signals and high-pass filtering of the gyroscope signals. The overall sensor fusion and filtering looks like this in Figure A.4.

Iterative Approach

The first iteration of the user interface had icons placed vertically with little space between each that had a start and an end position. The interface was not circular i.e. scrolling to the bottom of the user interface was possible. Upon registering a scroll amount generated by the smartwatch the user interface would scroll the webpage. A limitation did not allow users to scroll a desirable selection as the access time was higher. In order to tackle this problem, we designed a vertical circular carousel.

The circular carousel would repeat the sequence of the menu even after it reached the last element in the menu. The most appreciated feature of this design was that there was no boundary for the item list on the menu. We incorporated 5 different application simulations into this menu design.

Appendix B

User Study and Analysis

The user study and analysis appendix explains the various intermediate steps such as procedures and proofs taken during the study and analysis. This bridges the gaps where interpretation could be incomplete.

B.1 Experiment Procedure

The experiment design consisted of 4 parts.

- User
- Application
- Procedure
- Analysis

Here, we focus on the procedure part. The users were asked to perform each task consisting of 9 targets, 3 times with 3 different indexes of difficulty with a collective 12 users. In total there were $9 \times 4 \times 3 \times \times 3 \times 12 = 3888$ conditions. The users were asked to fill out a questionnaire following the experiment.

Questionnaire

The questionnaires were framed to determine the user's preferred interaction technique. The data obtained from each of the interaction techniques could contradict the experience the users had. This was the reason for a questionnaire.

The questionnaire was of 2 pages. The first page was framed to understand which modality the users preferred and the cognitive strategies for working with each technique. The second page revealed the user preference scores based on confidence and difficulty. The users were asked to rate their confidence using the technique to navigate, control the cursor and selecting the target by positioning the cursor at the center of the target. The questionnaire is as shown below.

B.2 Statistical Analysis

The statistical analysis on the data was run to determine the significant effects within the independent variables and between the levels of each independent variable. The various tests conducted were

1. 2-way repeated measure for all continuous data where ID, Technique and Block as independent variables.
2. Pairwise t-tests for determining the effect within each level of the independent variable.
3. Cochran Q-test for dichotomous data i.e. Overshooting.
4. McNemer's test for determining the effect of individual levels of independent variables.

USER_ID: ______

Questionnaire

1. In which task did you find yourself to be most confident while performing target acquisition?

2. Which task/s did you feel more difficult to complete? (1-4)

3. Did you feel you required more number of trials after starting the experiment tasks? if yes in which task? (1-4)

4. How did you manage to control the cursor speed with relation to the task 3 (Gesture as coarse pointing and Touch as fine pointing)?

5. In which task/s you wanted to rest more?

6. In which of the tasks, you felt you had already reached the target but required more repositioning?

7. What type of strategy did you employ to reach small targets that were far apart in all the tasks (if any) ?

1

Figure B.1

USER_ID: ______

8. Did you find the cursor movement easy in all tasks? If not in which task/s and what were the factors that you found affecting the speed of the cursor movement?

9. **For the following questions please answer from 0 to 5 where 0 is no difficulty and 5 is most difficult:**
   - Difficulty of Task 1 ________
   - Difficulty of Task 2 ________
   - Difficulty of Task 3 ________
   - Difficulty of Task 4 ________

10. **For the following questions please answer from 0 to 10 where 0 is no confidence and 10 is most confident**
    - Confidence in Task 1: ________
    - Confidence in Task 2: ________
    - Confidence in Task 3: ________
    - Confidence in Task 4: ________

2

Figure B.2

B.3 User Comments

1. *"When my arm was at the extreme extent, I switched to touch."* — $T01_A$ Symphony:
2. *"The touch movements was little iffy due to the position of the arm."*
3. *"The coarse pointing in gesture took time to adjust"* — $T01_A$ Gesture:
4. *"I recommend to remove the "slow-down" feature to make the technique more robust."* — $T02_A$ Symphony:
5. *"Slowing down of cursor in gesture caused troubles to acquire the target. It caused overshoot when I went to select the target."* — $T02_A$ Gesture:
6. *"The touch panel size is smaller so the cursor needs more acceleration"* — $T03_A$ Touch:
7. *"The smartwatch orientation change is difficult to adapt while using."* — $T03_A$ Gesture:
8. *"Replacing the watch with an ergonomic controller such as the Wii-Controller could be better for gesture recognition."*
9. *"Moving cursor should be boundless, moving it from one side could show up on the other side."*
10. *"I felt like the cursor was not moving in the direction I expected it to move."* — $T04_A$ Gesture:
11. *"I eventually needed more reposition of the cursor to select the target."* — $T04_A$ Base Controller:
12. *"I found the cursor movement to be difficult in during fine pointing using gesture."* — $T07_A$ Gesture:
13. *"The glasses got warm after a while, it was disurbing"* — $T08_A$:
14. *"I don't like tapping on the screen while performing the movement in gesture. This way I need to move both my arms."* — $T08_A$ Gesture:
15. *"I found using base controller, the cursor movement from the right to left side of the screen was difficult."* — $T09_A$ Base Controller:

Bibliography

Youngseok Ahn, Sungjae Hwang, HyunGook Yoon, Junghyeon Gim, and Jung-hee Ryu. Bandsense: Pressure-sensitive multi-touch interaction on a wristband. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pages 251–254. ACM, 2015.

Jarvis Trey J Arthur, Randall E Bailey, Steven P Williams, Lawrence J Prinzel, Kevin J Shelton, Denise R Jones, and Vincent Houston. A review of head-worn display research at nasa langley research center. In *SPIE Defense+ Security*, pages 94700W–94700W. International Society for Optics and Photonics, 2015.

Daniel Ashbrook, Kent Lyons, and Thad Starner. An investigation into round touchscreen wristwatch interaction. In *Proceedings of the 10th international conference on Human computer interaction with mobile devices and services*, pages 311–314. ACM, 2008.

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

Gerald Bieber, Thomas Kirste, and Bodo Urban. Ambient interaction by smart watches. In *Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '12, pages 39:1–39:6, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1300-1. doi: 10.1145/2413097.2413147. URL `http://doi.acm.org/10.1145/2413097.2413147`.

Renaud Blanch, Yves Guiard, and Michel Beaudouin-

Lafon. Semantic pointing: improving target acquisition with control-display ratio adaptation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 519–526. ACM, 2004.

Gabor Blasko and Steven Feiner. An interaction system for watch computers using tactile guidance and bidirectional segmented strokes. In *Wearable Computers, 2004. ISWC 2004. Eighth International Symposium on*, volume 1, pages 120–123. IEEE, 2004.

Costas Boletsis, Simon McCallum, and Brynjar Fowels Landmark. The use of smartwatches for health monitoring in home-based dementia care. In *Human Aspects of IT for the Aged Population. Design for Everyday Life*, pages 15–26. Springer, 2015.

Richard A. Bolt. “put-that-there”: Voice and gesture at the graphics interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '80, pages 262–270, New York, NY, USA, 1980. ACM. ISBN 0-89791-021-4. doi: 10.1145/800250.807503. URL `http://doi.acm.org/10.1145/800250.807503`.

Andrew Bragdon, Rob DeLine, Ken Hinckley, and Meredith Ringel Morris. Code space: touch+ air gesture hybrid interactions for supporting developer meetings. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, pages 212–221. ACM, 2011.

Rahul Budhiraja, Gun A Lee, and Mark Billinghurst. Using a hhd with a hmd for mobile ar interaction. In *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*, pages 1–6. IEEE, 2013a.

Rohit Budhiraja, Gun A Lee, and Mark Billinghurst. Interaction techniques for hmd-hhd hybrid ar systems. In *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*, pages 243–244. IEEE, 2013b.

Xiang'Anthony' Chen, Tovi Grossman, Daniel J Wigdor, and George Fitzmaurice. Duet: exploring joint interactions on a smart phone and a smart watch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 159–168. ACM, 2014.

Kwangsu Cho, Ju-Hwan Lee, Byeong-Taek Lee, and Eunil Park. Effects of feedforward in in-air remote pointing. *International Journal of Human-Computer Interaction*, 31(2): 89–100, 2015.

Andrea Colaço, Ahmed Kirmani, Hye Soo Yang, Nan-Wei Gong, Chris Schmandt, and Vivek K Goyal. Mime: compact, low power 3d gesture sensing for interaction with head mounted displays. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, pages 227–236. ACM, 2013.

Artem Dementyev and Joseph A Paradiso. Wristflex: Low-power gesture input with wrist-worn pressure sensors. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pages 161–166. ACM, 2014.

Alan Dix. *Human-computer interaction*. Springer, 2009.

Sarah A Douglas and Anant Kartik Mithal. The effect of reducing homing time on the speed of a finger-controlled isometric pointing device. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 411–416. ACM, 1994.

Jon Froehlich, Jacob O. Wobbrock, and Shaun K. Kane. Barrier pointing: Using physical edges to assist target acquisition on mobile device touch screens. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility*, Assets '07, pages 19–26, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-573-1. doi: 10.1145/1296843.1296849. URL `http://doi.acm.org/10.1145/1296843.1296849`.

Rui Fukui, Naoki Hayakawa, Masahiko Watanabe, Hitoshi Azumi, and Masayuki Nakao. Hand Gesture Interface for Content Browse Using Wearable Wrist Contour Measuring Device. pages 1222 – 1229, 2015. ISBN 9781479999934.

Jérémie Gilliot, Géry Casiez, and Nicolas Roussel. Impact of form factors and input conditions on absolute indirect-touch pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 723–732. ACM, 2014.

Jens Grubert, Matthias Heinisch, Aaron Quigley, and Dieter Schmalstieg. Multifi: Multi fidelity interaction with displays on and around the body. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3933–3942. ACM, 2015.

Steven J Henderson and Steven Feiner. Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret. In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*, pages 135–144. IEEE, 2009.

Juan David Hincapié-Ramos, Xiang Guo, Paymahn Moghadasian, and Pourang Irani. Consumed endurance: A metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 1063–1072. ACM, 2014.

Steven Houben and Nicolai Marquardt. Watchconnect: A toolkit for prototyping smartwatch-centric cross-device applications. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1247–1256. ACM, 2015.

Barry Kollee, Sven Kratz, and Anthony Dunnigan. Exploring gestural interaction in smart spaces using head mounted devices with ego-centric sensing. In *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction*, SUI '14, pages 40–49, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2820-3. doi: 10.1145/2659766.2659781. URL `http://doi.acm.org/10.1145/2659766.2659781`.

Werner A König, Jens Gerken, Stefan Dierdorf, and Harald Reiterer. Adaptive pointing–design and evaluation of a precision enhancing technique for absolute pointing devices. In *Human-Computer Interaction–INTERACT 2009*, pages 658–671. Springer, 2009.

Gierad Laput, Robert Xiao, Xiang 'Anthony' Chen, Scott E. Hudson, and Chris Harrison. Skin buttons: Cheap, small, low-powered and clickable fixed-icon laser projectors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pages 389–394, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-3069-5. doi: 10.1145/2642918.2647356. URL http://doi.acm.org/10.1145/2642918.2647356.

I Scott MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Human-computer interaction*, 7(1):91–139, 1992.

David C McCallum and Pourang Irani. Arc-pad: absolute+relative cursor positioning for large displays with a mobile touchscreen. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 153–156. ACM, 2009.

Anne Collins McLaughlin, Wendy A Rogers, and Arthur D Fisk. Using direct and indirect input devices: Attention demands and age-related differences. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 16(1):2, 2009.

Christiane Moser and Manfred Tscheligi. Physics-based gaming: exploring touch vs. mid-air gesture input. In *Proceedings of the 14th International Conference on Interaction Design and Children*, pages 291–294. ACM, 2015.

Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P Irani, and Michel Beaudouin-Lafon. High-precision pointing on large wall displays using small handheld devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 831–840. ACM, 2013.

Stephen Oney, Chris Harrison, Amy Ogan, and Jason Wiese. Zoomboard: A diminutive qwerty soft keyboard using iterative zooming for ultra-small devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2799–2802, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2481387. URL http://doi.acm.org/10.1145/2470654.2481387.

Sharon Oviatt, Rachel Coulston, and Rebecca Lunsford. When do we interact multimodally?: Cognitive load and multimodal communication patterns. In *Proceedings of the 6th International Conference on Multimodal Interfaces*, ICMI '04, pages 129–136, New York, NY, USA, 2004. ACM. ISBN 1-58113-995-0. doi: 10.1145/

1027933.1027957. URL `http://doi.acm.org/10.1145/1027933.1027957`.

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

Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. User-defined gestures for augmented reality. In *Human-Computer Interaction–INTERACT 2013*, pages 282–299. Springer, 2013.

Lorenzo Porzi, Stefano Messelodi, Carla Mara Modena, and Elisa Ricci. A smart watch-based gesture recognition system for assisting people with visual impairments. In *Proceedings of the 3rd ACM international workshop on Interactive multimedia on mobile & portable devices*, pages 19–24. ACM, 2013.

M. T. Raghunath and Chandra Narayanaswami. User interfaces for applications on a wrist watch. *Personal Ubiquitous Comput.*, 6(1):17–30, January 2002. ISSN 1617-4909. doi: 10.1007/s007790200002. URL `http://dx.doi.org/10.1007/s007790200002`.

Holger Regenbrecht, Jonny Collins, and Simon Hoermann. A leap-supported, hybrid ar interface approach. In *Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration*, pages 281–284. ACM, 2013.

Gerhard Reitmayr and Dieter Schmalstieg. Mobile collaborative augmented reality. In *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*, pages 114–123. IEEE, 2001.

Jun Rekimoto. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *Wearable Computers, 2001. Proceedings. Fifth International Symposium on*, pages 21–27. IEEE, 2001.

Marcos Serrano, Barrett Ens, Xing-Dong Yang, and Pourang Irani. Gluey: Developing a head-worn display interface to unify the interaction experience in distributed display environments. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 161–171. ACM, 2015.

Katie A Siek, Yvonne Rogers, and Kay H Connelly. Fat finger worries: how older and younger users physically interact with pdas. In *Human-Computer Interaction-INTERACT 2005*, pages 267–280. Springer, 2005.

DWF Van Krevelen and R Poelman. A survey of augmented reality technologies, applications and limitations. *International Journal of Virtual Reality*, 9(2):1, 2010.

Wouter Van Vlaenderen, Jens Brulmans, Jo Vermeulen, and Johannes Schöning. Watchme: A novel input method combining a smartwatch and bimanual interaction. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, pages 2091–2095, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3146-3. doi: 10.1145/2702613.2732789. URL `http://doi.acm.org/10.1145/2702613.2732789`.

Jacob O Wobbrock, Edward Cutrell, Susumu Harada, and I Scott MacKenzie. An error model for pointing based on fitts' law. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1613–1622. ACM, 2008.

Jianming Wu and Tomonori Yazaki. Infofinder: Just-in-time information interface from the combination of an hwd with a smartwatch. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*, MUM '15, pages 355–359, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3605-5. doi: 10.1145/2836041.2841207. URL `http://doi.acm.org/10.1145/2836041.2841207`.

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

Chao Xu, Parth H Pathak, and Prasant Mohapatra. Finger-writing with smartwatch: A case for finger and hand gesture recognition using smartwatch. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, pages 9–14. ACM, 2015.

Index

Typeset March 24, 2016