CountMarks: Multi-Finger Marking Menus for Mobile Interaction with Head-Mounted Displays

by

Jordan Pollock

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in partial fulfillment of the requirements for the degree of

Master of Applied Science

in

Human Computer Interaction

Carleton University

Ottawa, Ontario

Abstract

Head-mounted displays (HMDs) are becoming thinner, lighter and wireless. Soon we may see these displays used in public in devices like smart glasses. In this thesis, we designed, implemented and evaluated a novel multi-touch marking menu technique for use with HMDs. CountMarks extends conventional marking menus (gesture-based radial menus) by using multi-finger input on a mobile phone screen. This supports selecting items from each of four menus (one for each finger) with a single swipe, reducing the need for deeper menu hierarchies. We discuss the design of two variations of CountMarks, exploring selection efficiency, public acceptability, and ergonomic comfort. We conduct two studies: the first compares CountMarks to a traditional marking menu and finds one variation of CountMarks makes faster selections and allows for better search accuracy with only a small reduction in selection accuracy. Our second study evaluates CountMarks while standing and walking and with interaction occurring on hand-held and leg-mounted devices. Our results show that CountMarks can be used in the hand while standing or walking, and we confirm the difficulties with leg interaction. We evaluate the types of errors made by participants to suggest improvements to CountMarks as a whole and for leg interaction in particular. Finally, we present an application demonstrating the implementation of CountMarks in an existing user interface and we suggest directions for future work.

Acknowledgements

I first and foremost want to thank my supervisor Rob Teather. Without him – his trust, and supervision - I would not be here today. Dr. Teather gave me the freedom to explore my own interests wherever they took me (within reason) while pushing me towards excellence. So thank you Rob, I truly appreciate you taking me into your lab and for helping me refine all my awful drafts and ideas into something I can be proud of.

Of course, much of the efforts to help me complete this thesis took place behind the scenes with all the support from friends and family. I want to thank both my moms, Sheryl and Lisa who encouraged me to follow my passion for tech and who offered unconditional support and encouragement every step of the way. I want to thank my sister Jess who sent me cat pictures when I needed them most, and I want to thank two dear friends Bridgette and Celeste who I looked forward to seeing each week and whose visits helped push me through even the busiest of weeks. Finally, I want to say thanks to my friends both in Carleton and abroad who I could always talk to, day or night; who would check in on me when I would disappear into my work and relax with me when I needed a break. Special thanks go to those friends who volunteered their time and energy to helping me pilot test my experiments. Seriously, that was a life saver.

I have been shown a lot of kindness over the last few years and while I cannot possibly express how grateful I am for it all, I hope this is a start. I am very fortunate to have all of you in my life and I couldn’t have done this without you all.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

As predicted by Moore's law, computers are getting progressively smaller [1]. Computational tasks that used to require a room full of stationary computing machines are now done on a smartphone that fits in our pockets. As computer chips and batteries became smaller and lighter, computer displays similarly became thinner and lighter. When Ivan Sutherland introduced the "Sword of Damocles" head-mounted display (HMD) in 1968 [2] it was suspended from the ceiling, required two hands to hold and operate and was only able to display wireframes of simple shapes. Today, the Microsoft HoloLens[1] can process and display dynamic high-resolution augmented reality (AR) without external computation and weighs only 1.28 pounds. With the HoloLens and the recently announced Oculus Quest[2], a stand-alone virtual reality (VR) headset, the future of HMDs sees devices getting thinner, more comfortable and wireless to support everyday use away from a desktop computer.

The HoloLens and the most recent version of Google Glass[3] are not aimed at everyday consumers, but at industry workers. These devices serve as tools to provide workers with annotated images and instructions on how to perform their work tasks. Companies such as North[4] are pushing past industrial uses of HMDs by designing glasses that closely resemble regular everyday eyewear (Figure 1). This thesis assumes a future where mobile computing has moved from displays on smartphones to displays on glasses

[1] https://www.microsoft.com/en-CA/hololens
[2] https://www.oculus.com/quest/
[3] https://www.x.company/glass/
[4] https://www.bynorth.com/

that are worn casually in public. We refer to 'smart glasses' as an ideal HMD that looks indistinguishable from regular eyewear, but that performs the majority of functions smartphones currently do. Similar to how touch interaction is ubiquitous with smartphones, we explore interaction with HMDs where directly touching the display becomes less appropriate.

Figure 1: Focals by North look like regular glasses except for the bulkier sides

Recent surveys on smartphone app usage show that over half of time spent using smartphones is on music, multimedia, and other entertainment and communication apps[5]. If smart glasses follow as the evolution of smartphones, they will likely be designed for similar purposes, to access and communicate information while on the move in everyday life. For example, when running errands your grocery list may be ever-present in the corner of your smart glasses leaving your hands free to push a shopping cart and grab items from a shelf. At the same time, text and email notifications may pop up that can be

[5] http://www.businessofapps.com/data/app-statistics/

dismissed without interrupting or looking away from your task. The consequence of moving this information from smartphones to smart glasses is that the user no longer has the convenience of direct touch interaction to select and manipulate data. Instead, current methods for interacting with HMDs include the use of hand-held devices, such as with the Oculus Rift, as well as speech and gesture recognition, such as with the HoloLens. The latter two are impractical because speech recognition is inappropriate for use in public [3] and in air gestures are prone to arm fatigue [4] and inaccurate detection [5]. Handheld devices (e.g. gaming controllers) are more socially acceptable but can be expensive and cumbersome to carry around.

Interaction techniques must also be designed for the environment they will be used in. For example, mobile texting apps were not designed to be used while walking or driving. The consequence of this is that 3,450 people died and 391,000 were injured from distracted driving accidents in the USA in 2016[6] and distracted walking from cell phones is sending more people to the emergency room each year[7]. The use of HMDs may exacerbate this problem as HMDs can provide ever-present, information-rich displays that can be used while navigating through everyday life. Any proposed interaction technique for these devices must consider its use in everyday scenarios such as when sitting, standing and walking to understand the real-world implications of its use. Ideally, this technology should supplement the user's tasks instead of distracting them, and do so in a way that is efficient, ergonomic and disappears into the world [6]. Additionally, such

[6] https://www.nhtsa.gov/risky-driving/distracted-driving
[7] https://news.osu.edu/distracted-walking-injuries-soar-for-pedestrians-on-phones/

a technique should be socially acceptable, accurate and take advantage of existing technology the user may already own. To satisfy these requirements we first present and then empirically evaluate CountMarks, a multi-touch marking menu interaction technique that supports selection for mobile HMDs.

CountMarks extends marking menus [7] by employing multi-touch interaction, supporting up to four finger input instead of the traditional one touch input. With traditional marking menus (Figure 2), the user puts a single finger or stylus on the screen, and a circular menu appears. The user then performs a swipe gesture in the direction of the desired menu option to activate its selection. This selection could open a second marking menu in the direction of the swipe gesture and could continue this way as menu depth increases.

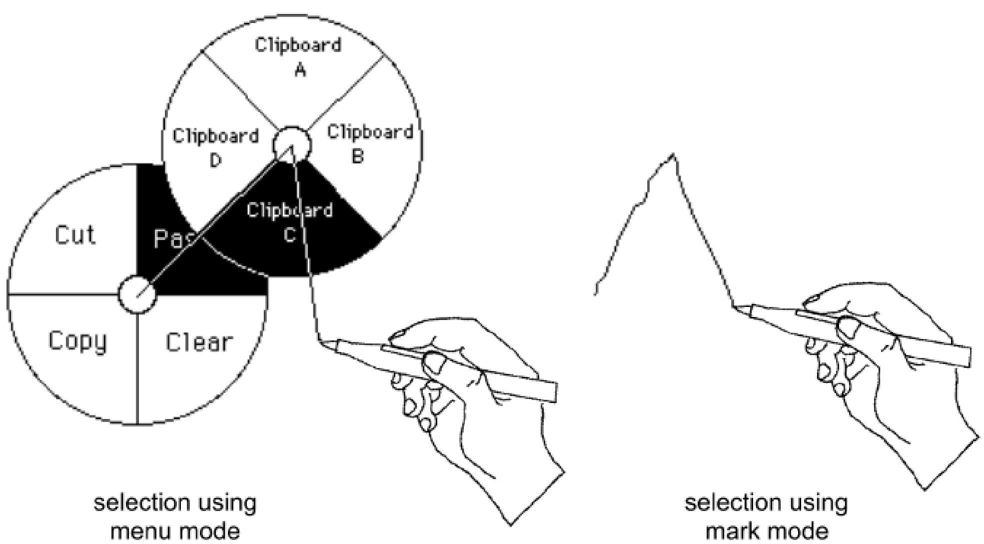

Figure 2: Hierarchical marking menu interaction by Kurtenbach (1993) [7]**. Users can either move through one menu at a time (left) or perform a single compound mark (right) to make a selection.**

CountMarks enables the selection of an individual menu item from multiple groups, based on the number of fingers touching the screen (Figure 3). While any multi-touch surface can be used with this technique, we implement CountMarks on a smartphone as the input device in our experiments; we speculate that the ubiquity of smartphones may make them ideal input devices for future HMD-based wearable systems (VR, AR, or smart glasses).

Figure 3: A demonstration of CountMarks. From left to right showing a) a user places two fingers on the screen to open the 2nd menu. b) Swiping right with two fingers to select the target

Numerous studies as early as the 1980s have revealed that pie menus (and their evolution, marking menus) offer faster and more accurate selection than linear menus [7, 8]. Despite this, marking menus are rarely seen in commercial applications. They are found in some design tools (e.g., Autodesk's Maya, Figure 4) and in some video games (e.g., Assassin's Creed and Dead Island, Figure 5), where they are used for quick selection. The main limitation is that within a single menu level, marking menus typically only support a maximum of 8 options to minimize error rates [9]. But because of this limitation, designers must increase the depth of their menu hierarchies when implementing marking menus, which is shown to cause users to make slower and less accurate selections [10]. Deep menu hierarchies do not follow 3 of Nielson and Molich's design heuristics [11]: minimize the user's memory load, prevent errors and provide shortcuts. As such, they are often not ideal to be used in commercial applications.

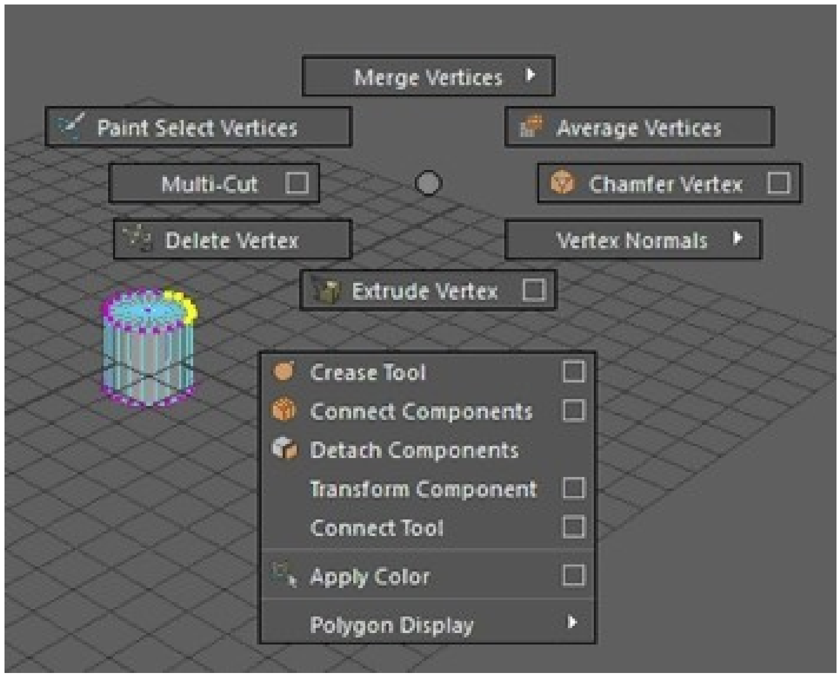

Figure 4: Marking menus in Maya, a 3D graphics application

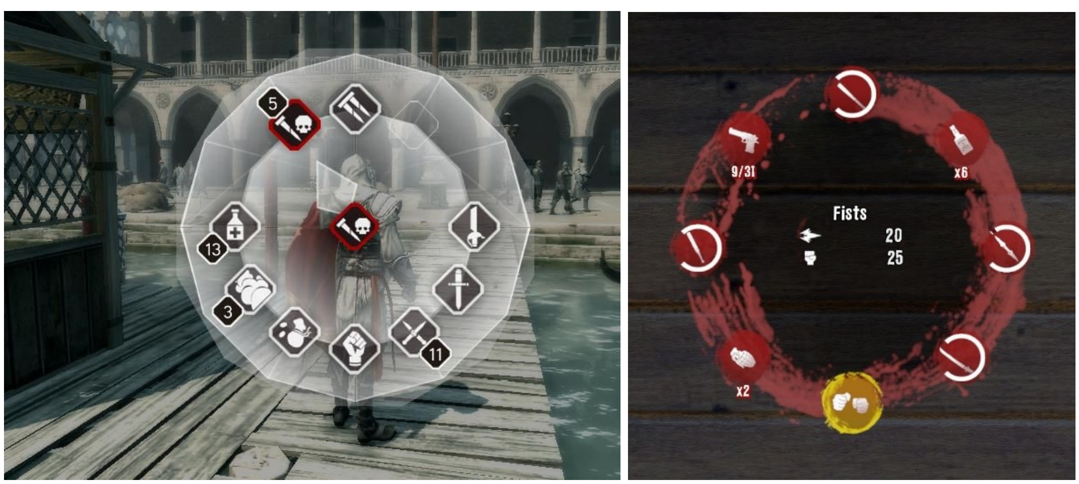

Figure 5: Weapon wheels in Assassins Creed: Brotherhood (left) and Dead Island (right)

1.1 Contribution

In this thesis, we present CountMarks, a newly designed mobile marking menu technique that allows for greater menu breadth than traditional marking menus by using multi-finger interactions. We first compare two variations of CountMarks to traditional marking menus and demonstrate the effectiveness of increased breadth in marking menu selection. We then assess and evaluate one of these variants and its ability to make quick and accurate menu selections in a variety of use mobile cases.

Our first study provides empirical evidence that CountMarks offers quicker selection and improved search accuracy over multi-stroke marking menus at the cost of a small decrease in selection accuracy.

Our second study compares the effectiveness of CountMarks while standing and walking, and while performing interactions with hand-held and leg-mounted devices. We provide the first (to our knowledge) evaluation of a marking menu interaction in these user stances and device positions. We demonstrate CountMarks' ability to be used successfully in handheld conditions while standing and walking, outperforming reported results from other marking menu variants. To improve upon the less accurate results for leg interactions, we analyze a selection of errors and provide suggestions for improving accuracy not only for CountMarks but mobile marking menus in general.

Finally, we contribute a proof of concept application for how CountMarks allows marking menu selection to be better implemented in future interfaces. We describe the strengths and weaknesses of using CountMarks in such interfaces and we suggest future work to improve its design.

1.2 Outline of the Thesis

This thesis is divided into 6 chapters. Chapter 2 presents a comprehensive literature review on marking menus and its variations, and on interaction techniques, including where and how interactions are performed.

In Chapter 3 we discuss the design and implementation of two variations of CountMarks. This chapter explores the limitations of marking menus and the justifications for why it is necessary to explore alternative implementations.

Chapter 4 presents an experiment evaluating the effectiveness of both CountMarks variants against traditional marking menus. Chapter 5 presents an experiment that explores using the higher performing variant of CountMarks in positions and movement cases. Both chapters include participants' information, procedure, apparatus, design, results, and discussions.

The final chapter summarizes findings and proposes design recommendations for the implementation of marking menu interaction with HMDs that are designed to be used in motion. We provide an example application of CountMarks and propose directions for future research related to marking menu interaction with mobile HMDs.

Chapter 2: Related Work

This thesis explores marking menus as a method for indirect touch input for mobile HMDs as an alternative to direct touch, cursor-based or in-air gesture-based approaches. Some of the most common interactions with smartphones are selection tasks such as choosing apps, selecting emails or interacting with information in social media feeds[8]. Interactions with smart glasses are likely to follow this trend. While a smartphone screen is a convenient location to perform direct touch selections and desktop computers use the stable 2D plane of the desk for precise mouse input, these are not suitable for smart glasses. The mobile nature of HMDs make it difficult to have a consistent plane for precise 2D input, and touch input around the face, while possible for HMDs [12, 13], would be impractical for smart glasses because the lenses should remain clean to see through. We instead look to menu-based interaction to support indirect interactions, rather than direct selection, specifically focusing on marking menus due to their known efficiency.

2.1 Marking Menus

Marking menus (see Figure 3) are swipe-gesture based menus introduced in the early 1990s by Kurtenbach [7] to create shortcuts for item selection. Marking menus have been consistently shown to be quicker and more accurate than typical linear menu selection

[8] http://www.businessofapps.com/data/app-statistics/

[14]. Due to the performance benefits of marking menus, and the extensive literature on linear menus [15], we do not consider linear menus further in our work.

Marking menus extend the concept of pie menus – circular menus that appear around the cursor when invoked [8], by allowing for compounded marks as shortcuts to items in deeper menu levels (Figure 2). Modern marking menus show that multiple individual strokes improve selection speed and accuracy [16]. The primary limitation of marking menus is that they only support up to 8 items per menu and two levels of depth to retain 90% selection accuracy [9]. This is a problem, since past research shows that broad menus (i.e., menus with more items in a single menu level) performs better than deep menus (i.e., menus requiring drilling down into a hierarchy, increasing the number of selections required to select an item). This is likely because users tend to get lost in deep menu hierarchies [17]. As a result, variations of marking menus have been created to increase breadth: the flower menu [18] (Figure 6a) which uses curved instead of linear strokes, and the zone and polygon menus [19] (Figure 6b and c) which let users make different selections based on where their swipe gesture begins. These variations demonstrate that increased breadth can increase selection speed and/or accuracy. Increasing breadth by having "overflow" items (those that do not fit in the 8 item marking menu) sit underneath in a linear menu was proposed in a patent by Kurtenbach [20] but does not appear to have been formally evaluated.

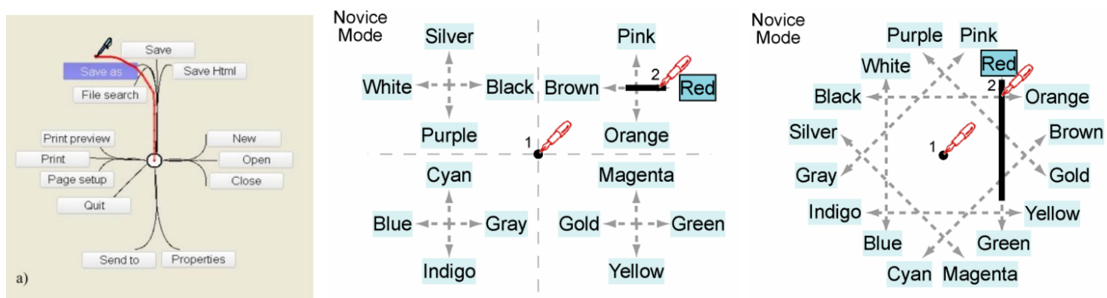

Figure 6: Marking menu variations that increase menu breadth - from left to right: a) Flower Menu (Bailly et al. 2008 [18] **b) Zone Menu and c) Polygon Menu (Zhao et al. 2006** [19]**)**

Perhaps the most closely related project to ours, Lepinski et al. [21] first explored multi-touch input for marking menus as a way to increase menu breadth. By holding different combinations of fingers (chords) to a touch surface different menus popup which the user can swipe on to make selections on a large touch surface (Figure 7a). Chording gestures are recognized with a camera near the touch surface. Most chords were found to be difficult to perform, but a simple set of chords (Figure 7b) were shown to make faster selections than modern multi-stroke marking menus.

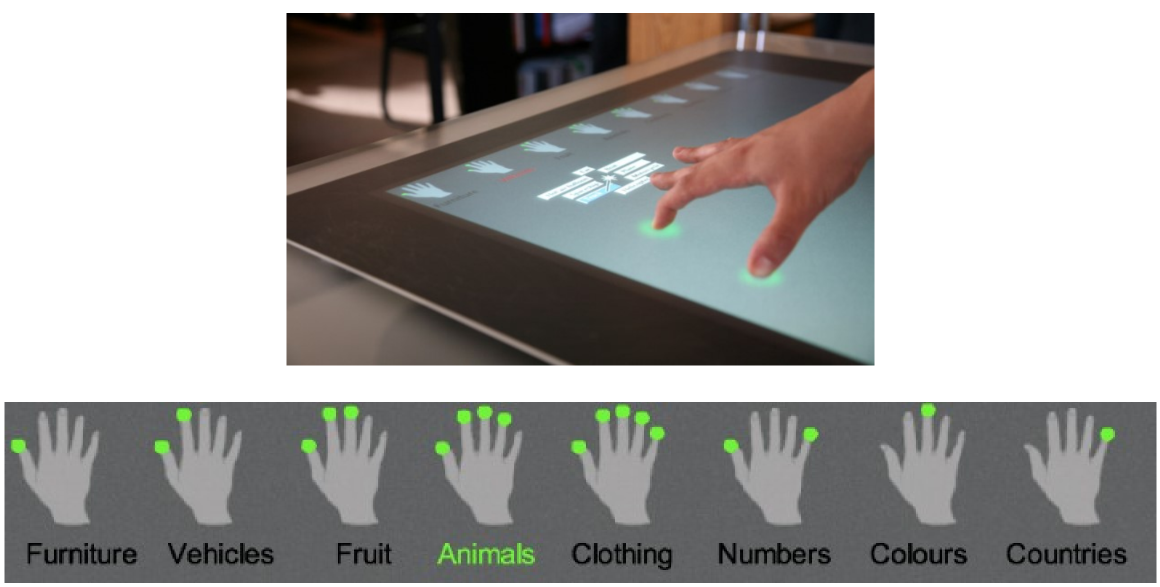

Figure 7: A multitouch marking menu user creating a chord gesture to open a marking menu (top). The set of chords used (bottom) (Lepinski et al. 2010 [21]**).**

Finger Count menus [12] use simpler chording gestures that let users select menus and items according to the number of fingers used in a gesture, regardless of which fingers are used. This technique uses two hands: one hand to select one of 5 menus and the other hand to select one of 5 items from that menu (Figure 8). Users can perform this type of gesture in the air [22, 23] or on touch surfaces [24]. On touch surfaces, Count Menus were found to be simple to use and easy to learn, although it was recommended that the thumb not be used due to ergonomic concerns of it being shorter than the other fingers. In-air Finger Count gestures were found to be faster than other in-air gesture-based menu selection techniques and twice as fast as 3D marking menus [25] while maintaining similar accuracy [23].

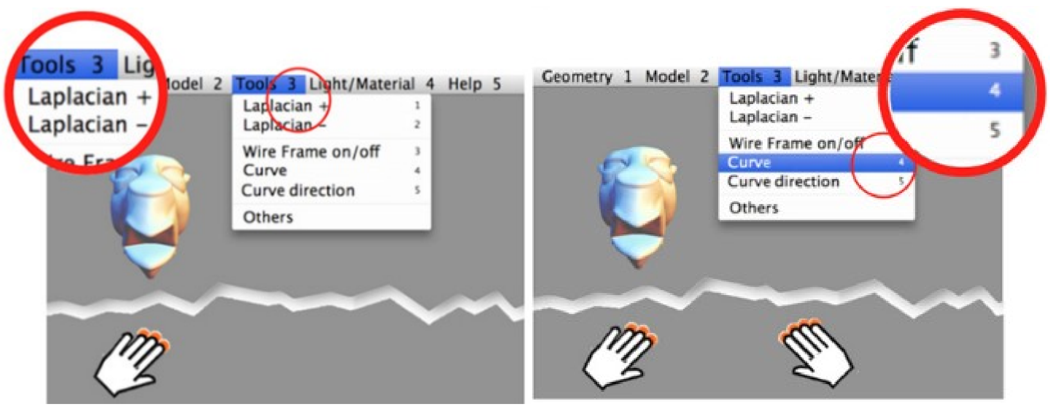

Figure 8: Count Menu (Bailly et al. 2012 [24]**). Selecting a menu (left) and an item (right).**

Our technique, CountMarks, employs Count Menus to simplify the chording structure of multi-touch marking menus to create an easy to use marking menu selection technique with expanded breadth. This thesis chooses to compare CountMarks to multi-stroke marking menus [26] because this variation is shown to be faster and more accurate than Kurtenbach's hierarchical marking menu.

2.2 Novice vs Expert Menu Selection

A major advantage of marking menus is that they seamlessly transition the user from Novice to Expert selection [7]. Most casual software users perform what would be considered "Novice" menu selections. Novice selection occurs when the user has a task to accomplish but may not be sure of how or where to find a menu item to accomplish that task. In graphical user interfaces, this involves the user searching through different menus until they find their target. Shneidermann [27] partially credits the success of the graphical user interface to its Novice appeal of "see and point versus learn and remember", when comparing the recognition of graphical user interfaces to the memory recall needed for command line interfaces.

In contrast, more experienced users may already know how to access a desired target item and either maneuver through the menu to select it without requiring search, or otherwise use shortcuts provided by the system. An example of Expert selection that most people use every day is the keyboard command Ctrl + V for pasting a copied item onto a page. If memorized, this is much quicker than using the mouse to navigate to the Edit menu and selecting the "paste" item, but the shortcut requires concerted learning and practicing efforts before it is quick to use. Due in part to this effort, these Expert tools are comparatively rarely used [28]. This is true even for experienced users who know that there are quicker tools available for selection [29]. Cockburn describes this as "being trapped in beginner mode" [30], where the user's familiarity with their current technique makes it more difficult to want to learn a more efficient one. When Kurtenbach developed marking menus [7], he aimed to eliminate this learning curve; users train themselves on the Expert commands without realizing it. With marking menus, the same series of swiping gestures used to navigate through a Novice menu are used for the Expert command. The challenge is that the user must first learn this new menu organization.

Depending on the type of menu there may be challenges at different levels of learning which can be addressed in different ways. Findlater et al. [31] simplified the Novice learning of menus by showing the most relevant items first and having the rest slowly fade in to help the user identify the few likely relevant targets. OctoPocus [32] focused on transitioning Novices into Experts by offering the user semi-transparent stroke suggestions to facilitating the learning of complex swipe gestures. The psychology and learning theories behind how users move from Novices to Experts in selection is out of the scope of this thesis but is detailed elsewhere [30].

2.3 Interaction Techniques

In-air gestures are the obvious choice for interacting with HMDs because they offer the freedom to interact with *all* of the space in front of the user at any time. This "anywhere, anytime" convenience is desirable for any interaction technique where users may be walking or standing. However, interaction techniques designed for public use must consider *social acceptability*. For an interaction technique to be socially acceptable the user should be able to operate it with "subtle and intuitive gestures" that do not attract attention from their surroundings [33]. Good examples are gestures that look or feel similar to everyday actions, and interactions that are similar to existing technologies [34]. In-air gestures can be obtrusive, attention-grabbing and often unnatural or dissimilar to normal gestures which can make them potentially embarrassing to use in public settings [13, 35, 36]. Additionally, in-air gestures have the well-known problem of "gorilla arm syndrome" where gesture can yield excessive arm fatigue [4].

To obtain the benefits of free-hand "anywhere, anytime" interaction without the potential side effects of fatigue, researchers have explored mounting depth trackers to the shoulder [37] or hip [38] to turn normal surfaces into touch surfaces or allow for in air gestures without raising the arm. Instead of optical depth trackers, other approaches use wearable capacitive devices to detect touch contact [39, 40]. These touch recognition techniques provide the benefit of haptic (touch) feedback to the user which is shown to increase gesture recognition accuracy and precision [41]. Aside from touch input on external devices, several researchers have considered using touch on the body itself as an input source. There is a growing body of work on employing the user's skin specifically as an input source [42-45]. Skin makes an interesting surface for interaction due to the large number of gestures one could perform such as grab, pull, press, scratch shear, squeeze and twist gestures [46]. We will discuss variations below.

2.3.1 On hand input

A gesture elicitation study on skin input [3] indicates that the palm was the most desirable place for touch interactions, with 51% of users eliciting actions there. PalmGesture [47] showed that multi-stroke palm gestures can be captured with 90% accuracy using an infra-red camera and a laser-line projector (Figure 9). They also found that users can comfortably interact with their hand in a variety of orientations, but that users prefer to perform gestures on just the palm and avoid interacting with their fingers.

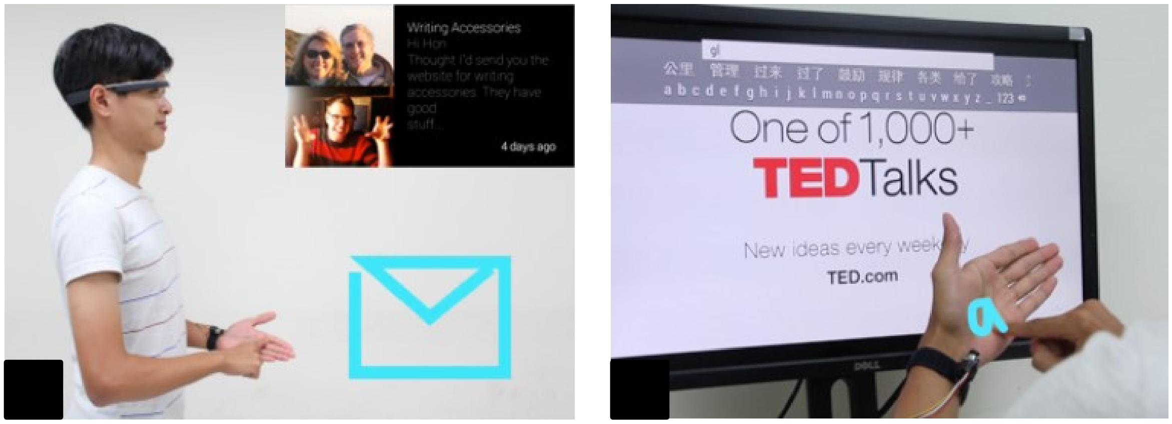

Figure 9: Examples of using PalmGesture for opening an email (left) or entering text for search (right) (Wang et al. 2015 [47]**)**

Other research explores using magnetic field detection to capture smaller, more precise between-finger input. With FingerPad [48], users perform gestures including writing numbers, scrolling, swiping and tapping. FingerPad was designed to offer users privacy in their interactions. Similarly, uTrack [49] detects thumb to finger interactions to enable 3D pointing, and NailO [50] uses finger interactions on the thumbnail for pressure and swipe gestures. Google's project Soli takes a different approach and uses Radar sensors to detect millimeter scale interactions between fingers [51]. Ring wearable devices are another popular field of research. Rings can take advantage of twisting gestures that wearers naturally perform on them [52], and can provide a convenient place to hold a depth sensing fish-eye lens camera for hand gesture recognition [53]. A full review of finger wearables is out of the scope for this thesis, for a more comprehensive look into finger wearables the reader is referred to Shilkrot et al.'s survey [54].

2.3.2 On body input

Most body interaction research focuses primarily on the forearm or back of the hand, but some look at input to the ears [55, 56]. SkinMarks [44] captures many such gestures by using rub-on tattoos with functional ink that can detect capacitive touch, squeeze and bend gestures anywhere on the body. Instead of pressure input, SkinTrack [45] uses electromagnetic waves to capture touch input on the skin near a smartwatch with up to 99% accuracy. Similarly, WatchSense [57] uses a forearm mounted depth camera to track both touch and mid-air gestures, capturing 3D touch and gesture input from the back of the hand with one or two fingers.

Despite 9% of all gestures from Tung et al's gesture elicitation study [3] being performed on the leg (compared to 51% on the hand, 12% on a ring, 10% between fingers), there is little to no research conducted on using the legs as places for input. Some researchers have looked at nearby areas. For example, Dobbelstein et al. [58] experimented with using belts as an input at the waist because "As the body's center of gravity, the hip is relatively steady while walking" and found it to support touch gestures in mobile settings. Additionally, methods for capturing input on a smartphone through fabric have been proposed [59] to allow for touch input to the outer thigh or wherever one has pockets. PocketMenu [60] takes this a step further and has users perform tap and swipe gestures through their pocket for manipulating a music application on a smartphone while in mobile settings. Tap and swipe gestures are performed at different heights on the phone in the pocket to determine which function is performed (Figure 10).

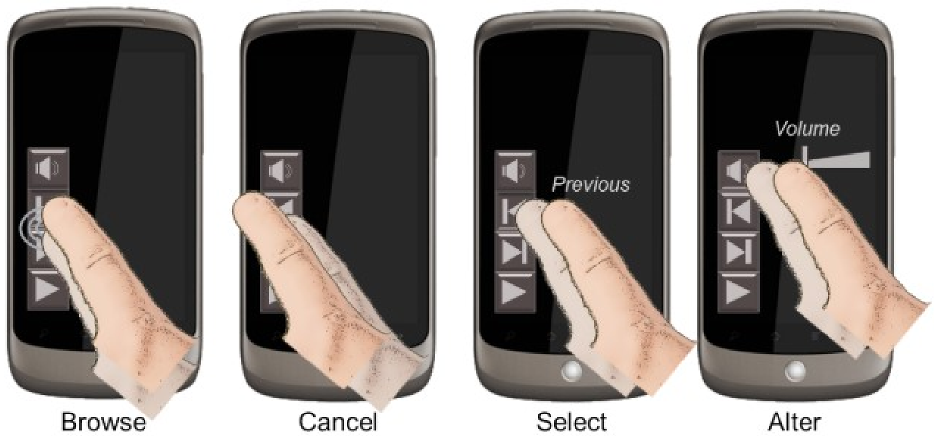

Figure 10: The four main gestures for PocketMenu (Pielot et al 2012 [60]**)**

Using everyday wearable fabrics and accessories for input have been of growing interest to many researchers. For accessories, Weigel & Steimle created DeformWear [61] a tiny nub that can take pressure, shearing (slide + push), and pinching gestures for input on rings, necklaces, and bracelets. Performance was worst with bracelets likely due to interactions occurring between two arms in motion. Yoon et al [62] created a single layer smart textile worn as a finger sleeve that can take bend and pressure input, demonstrating accuracy greater than 80% while walking, running and driving. For a full summary of smart textiles and fabrics, the reader is referred to Yoon et al.'s survey [62]. Whereas Dobbelsteain's belt prototype required the user to rest their hand on their pants pocket for comfortable interaction, and PocketTouch required the user to use corners of the phone for reference for their touch interactions, there is no research quantitatively examining reference-free leg interaction.

2.4 Mobile input

If the device is being used in public, it is likely the user may be mobile. Due to public safety concerns, any method for interacting with mobile technology should consider how interactions may be performed during different levels of mobility. While some interaction techniques made for wearable technology fail to be tested in mobile conditions [63], others test a range of scenarios such as walking [35, 60], running [56], simulated driving [62] and while grasping other objects [64]. Studies with NotifEye [35] and PocketMenu [60] both required participants walk down sidewalks to test their devices in the wild. NotifEye had participants view and dismiss floating semi-transparent notifications in an Epson Moverio BT-100 see-through interactive HMD. The study showed that participants were able to manage interacting with notifications using a small finger rub pad while navigating through a public street. PocketMenu similarly showed that users were able to pause and change music volume quickly and accurately on their phone through their pocket while walking.

Other researchers take more controlled approaches in studying mobile interaction by having participants use a treadmill. Skinput [43] is a device worn near the bicep that allows users to perform touch input on their forearm. Because it is acoustically driven, the researchers were sensitive that using it while walking and running could be unreliable due to environmental noise. They found that walking produced 100% reliable input for males (86.7% for females) but while jogging it dropped to 83.3% (60% for females), providing practical information for how their device performs in real-world use cases. When testing FingerPad [48] the researchers included a walking condition to evaluate how their private gesture system really performs. This walking condition yielded significantly more selection errors than standing. Wristwhirl [65] avoids the drawbacks from walking by using wrist input instead of touch. The authors found that arm movements while walking, are mostly independent of wrist movements. Because of this, users were able to draw "high-quality images" even while walking. Yoon et al's [62] smart textile finger sleeve takes mobility testing a step further and attempts to assess a full range of motion by including standing, walking and simulated driving tasks. They found that gripping a driving wheel created much more noise for hand input than input while standing or walking.

2.5 Summary

This thesis aims to expand the understanding of mobile interactions by evaluating CountMarks where the user is sitting, standing and walking. While marking menus have been designed for smartphone use [63], they have never been empirically evaluated in mobile use cases. We additionally aim to explore different locations of input, testing tabletop, hand and leg interaction locations. This requires us to simply and refine multi-touch marking menus [21] to be used on smaller touch surfaces that may be used in motion. While hands and arms have been extensively discussed in the literature above, we identified limited research on leg interactions. To our knowledge, this is the first evaluation of marking menu interaction while in motion and with input at the leg.

Chapter 3: Design and Implementation of CountMarks

In this chapter we describe CountMarks. CountMarks is a novel mobile interaction technique for HMDs based upon marking menu selection [7]. CountMarks expands on marking menus by adding a finger count component where the number of fingers placed on the touch surface allows for up to 4 times the number of selections possible in each swiping direction. It takes advantage of the speed, accuracy, and scale-invariant nature of marking menus and adds the simplicity and intuitive nature of Count Menus [24]. While Count Menus use both hands for interaction, we limit the current implementation of CountMarks to one hand for simplicity and practicality reasons. Furthermore, based on previous findings [24], we do not include the thumb in CountMarks interaction.

Marking menus can reliably hold up to only 8 menu items in a single menu level, or 64 items over two levels. In contrast, CountMarks can fit 32 items in a single menu level (four fingers, eight directions each) and 1024 (32^2) items over two menu levels. We use at most eight items on each submenu, corresponding to the eight cardinal directions and the upper limit on marking menus [7]. To support a larger number of menu items, we include a mode-changing technique whereby the user double taps the screen with one finger to access another set of menus or items. This effectively doubles the number of items in one level to 64 and up to 4096 (64^2) items across two levels. The mode change toggles the state of selectable options. This does not reset after each selection but instead can be toggled on and off depending on which menus or menu options the user wishes to interact with. For example, the regular set of swipe interactions could select from main content on an app, but a double tap could be used to begin selecting auxiliary settings visible in the same screen.

The primary advantage of CountMarks is that it supports a larger number of menu items in fewer sub-menu layers than marking menus. Previous work [17] has shown that hierarchical navigation with marking menus can be difficult because users can forget where they are in the hierarchy and can take longer to make a selection. More menu levels also require the user to rely on memory to recall where items are located, whereas broader menus allow users to merely recognize the item on the screen [27] . It is well known that people are better at recognition than memory recall.

We design CountMarks specifically with smart-glasses in mind, due to the challenges faced by their public and mobile nature. We describe these design considerations next.

3.1 Social acceptability

Because of the public nature of smart glasses, we designed CountMarks to comply with Hsieh et al.'s [33] recommendations for social acceptably:

1. Isolate sensing technology from the glasses,
2. Use relative pointing for adapting to various postures,
3. Design small movements for subtle interaction,
4. Aim for intuitive gestures,
5. Enhance tangibility.

CountMarks is designed to be subtle, intuitive and attract less attention from the user's surroundings by using small touch gestures on a peripheral surface. Noting the poor precision of in-air interaction due to the lack of haptic feedback [5], we employ CountMarks on a touch-sensitive surface. In this thesis, we implement CountMarks on a smartphone which allows users to take advantage of hardware that they already own and eliminates the need to buy and carry around an additional device. Future versions could instead be implemented on smart textiles or devices that allow for skin input. Swipe gestures on skin or clothing give users precise haptic feedback while allowing them to perform gestures wherever is most comfortable. We predict this may be on the hand or the upper leg.

Hsieh et al.'s recommendations were intended for improving in-air gestures so they suggest relative, not absolute, pointing for different postures. For example, instead of pointing directly at an object in a display (absolute pointing), the hand could control a cursor which moves in the same direction as the hand regardless of where the hand begins its movement (relative pointing). Since we are instead using touch interaction, we use relative swipe gestures in the form of marking menus. Marking menus allow the user to begin a swiping gesture at any point on a touch surface such that the detected movement is relative to where the user starts the interaction. This facilitates eyes-free interaction as the user does not need to see the surface because they can to begin a selection anywhere. It also means that gestures can be performed in any posture and swipe directions can remain relative to the surface of interaction. Finally, we believe that because swipe gestures are commonly used in laptops and smartphones, users will find marking menu interaction intuitive to perform.

We next discuss two proposed variations of CountMarks: MenuCount and ItemCount.

3.2 Variation 1: MenuCount

The MenuCount variation of CountMarks presents different marking menus depending on the number of fingers touching the screen. By holding their fingers on the screen, users can open and preview each menu. Releasing their fingers closes the menu. Adding or removing fingers will change which menu is displayed. For example, in Figure 11 the user has placed two fingers on the screen to activate the “Animals” menu.

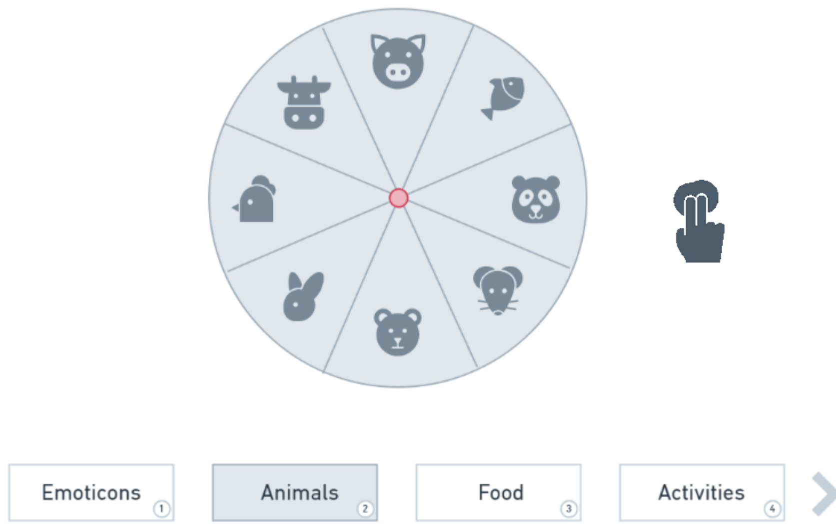

Figure 11: The user presses two fingers to the touch surface and the second menu opens with a cursor in the middle.

Note that the user does not need to touch the "Animals" tag but can place 2 fingers anywhere on the screen. Opening the menu highlights its tag. To make a selection, the user simply swipes in the direction of the desired menu item with the same number of fingers used to open the menu. Our current implementation supports up to four different menus (corresponding to up to four fingers contacting the screen), but if extra menus are needed then the aforementioned double-tap mode change can be used. As seen in Figure 11, the arrow icon on the right side of the menus indicates to the user that there is another set of menus available. After the mode change has been performed, the menus change and the arrow switches to the left to let the user know they can double tap again to return to the previous menus. This combination of finger-count and marking menu selection allows the user to choose from multiple menus and select their desired item in one fluid gesture. Due to the clear one-to-one mapping of menus to the fingers, and the limited amount of information shown at any given time, we anticipate this to be the more intuitive variation of CountMarks.

3.3 Variation 2: ItemCount

The second variant of CountMarks is ItemCount. ItemCount starts with a large radial menu containing up to eight linear sub-menus in each of (up to) eight directions. Each of these linear menus can display up to four individual items at once. These items can be selected by touching the screen with the appropriate number of fingers, and a swiping in the direction of that item's menu. Holding down fingers highlights all menu items selectable with that number of fingers; one finger highlights all top menu items, two fingers highlight all the second items, etc. The user can then swipe in the direction of the desired sub-menu to make their selection. For example, in Figure 12, if the desired menu item was "Chicken" (the 2nd item in the list on the left side of the radial menu) the user would hold two fingers to the touch surface and swipe left. We present only four items in each linear sub-menu due to the number of fingers available to make one-handed selections. Similar to the mode change used with MenuCount to show additional menus, ItemCount instead uses a mode change to show four additional items. The availability of the mode-change is depicted by ellipses at the bottom of each sub-menu seen in Figure 12. Once activated, the items change and the ellipses move to the top of the menus. We expect that due to the information density of displaying 32 items at once, users will have a more difficult time finding their target before making a selection.

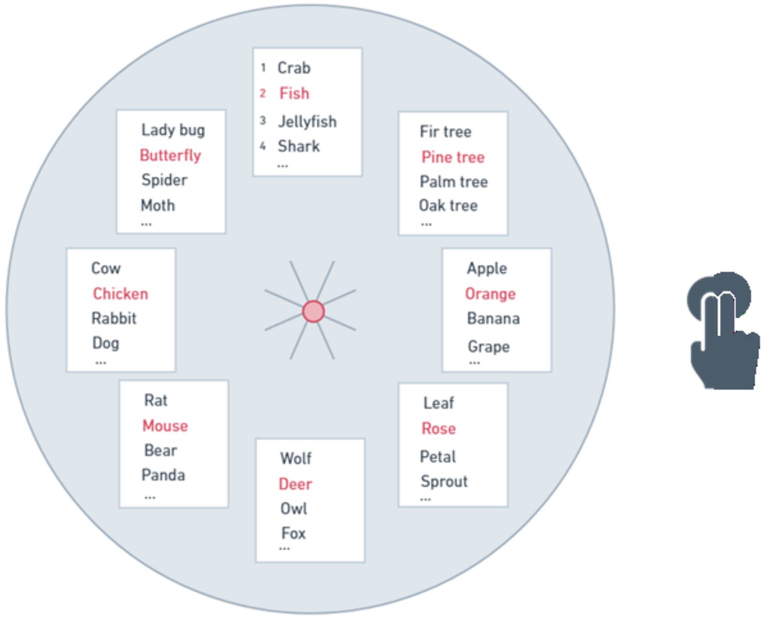

Figure 12: The user presses two fingers to the touch surface which highlights all the 2nd options for each sub-menu

Each variation of CountMarks is designed to be performed using the user's body as an interaction surface. We considered two locations: the hand and the leg which will be discussed below.

3.4 Location of Interaction

We anticipate that CountMarks will work best when performed on the human body because it is comfortable and easy for people to interact with [46]. Body interactions provide haptic and proprioceptive feedback that may allow for more precise and well-coordinated interactions. While capturing on-body interactions is an on-going engineering challenge [45, 46, 48], we implement our prototype on a smartphone to take advantage of its touch sensing capabilities. Smartphones are also a practical tool for implementation as they are already used and carried around by most people and may be a viable solution for interacting with HMDs until on-body sensing technology improves. With the ideal scenario being on-body interaction and the current practicality of a smartphone implementation, we look at two main locations for interactions: the palm and leg.

Palm interaction represents the most intuitive on-skin implementation since people naturally perform on-skin interactions with their palm [3]. Additionally, it is also the most natural place to interact with a smartphone. While we acknowledge that hand-held marking menu interactions requires users to use their phone in an unfamiliar way, we expect this is still a comfortable way to use CountMarks due to the amount of time people spend holding their phone daily. The leg is a less obvious but potentially ergonomic and efficient location for interaction. Notably, palm interactions require the use of both hands – the acting hand and the receiving hand. In a mobile setting where people are interacting with the environment as well as their devices, this is not desirable. It also requires both hands and arms to be held in front of the user which is tiring, much like in-air gestures. If a user is to interact with an HMD like smart glasses repeatedly, palm interaction will be impractical due to fatigue and potential for repetitive stress. With interaction occurring on the leg, however, the user can rest their arms at their side, as they naturally do while standing, and perform interactions where their hand meets their leg. This relaxed interaction may allow the user to use CountMarks without fatigue. This also gives the user a free hand to interact with the world around them or interact other possible devices. CountMarks could even be implemented on both legs, using two hands to double the number of fingers available for interaction. We note that although in this research we implement our prototype using a smartphone strapped to the user's thigh for leg interactions, we are not proposing that users actually strap a phone to their leg. Instead, this is an exploration of how CountMarks might be implemented when using smart textiles or a device that can detect touch input through fabric. While this thesis focuses on body interaction, we note that future work could explore implementing CountMarks on accessibility devices such as wheelchairs.

3.5 Limitations

The primary limitation of marking menus in general is that users must be taught how to use them. While other research has addressed learnability of marking menus [14], that is out of the scope of this thesis. We merely ensure the usability of CountMarks is on par with other marking menu variants. A limitation of CountMarks specifically is that it does not work when the touch surface is small and the display. After all, if a user placed more than one finger on a smartphone display, they would occlude most of the screen. We instead intend that CountMarks be used with HMDs like smart glasses. We also note that the current hand-held smartphone implementation of CountMarks requires both hands to perform interactions: one hand holds the smartphone, one hand to make swipe gestures. However, because CountMarks required mobility of all fingers, one-handed operation is not currently possible while holding the smartphone. Future work may explore different grips to allow for one-handed use. One-handed use also gives concern for scalability. While we will later demonstrate CountMarks in an example application, we will also discuss how 4 finger interaction will become difficult with more complicated UI. In these cases we argue that such a UI would not be suited for use in mobile setting and as such CountMarks may not be the best interaction method for the task. We believe CountMarks will be best suited for selection when there are 64 or fewer items available for selection at any one time. While CountMarks is capable of going into deeper menu structures, participants may struggle to navigate through more than one level of depth due to the exponential number of items potentially available (1 level = 64 items, 2 levels = 4096 items).

We ran two studies to evaluate our design of CountMarks that we will discuss next. The first compares the two versions of CountMarks against each other and a traditional marking menu. The second evaluates the different device positions and mobility conditions we described in this chapter.

Chapter 4: User Study 1

We conducted a user study to compare two variations of CountMarks against a traditional marking menu. Instead of Kurtenbach's original hierarchical marking menu [7], we compare CountMarks against the multi-stroke marking menu [26] (in this chapter referred to just as Marking Menu) due to its faster selection speed, improved selection accuracy and reduced UI space needed for menu depth. Additionally, the multi-stroke gestures are more similar to CountMarks gestures than the compound strokes from Kurtenbach's marking menus. The ethics protocol for this user study was reviewed by the Carleton University Research Ethics Board with CUREB-B clearance.

4.1 Participants

We recruited 18 participants. Their average age was 25.95 years (SD = 6.19 years), 10 were female, all were right-handed. Participants were recruited via posters around campus as well as through a post on a Carleton research Facebook page.

4.2 Apparatus

We used a Samsung Galaxy S8 smartphone (running Android 8.0) as the touch input device. The smartphone was centered horizontally and secured to the desk with Velcro 10 inches from the participant. A 23.5-in. BenQ 1920 x 1080p computer monitor was positioned 19 inches away from the participant (Figure 13). The participant interacted with a Unity Android 7.0 application that controlled the desktop Unity app. The desktop computer used an Intel Core i7-7700K CPU at 4.20GHz with 32 GB of RAM and 64-bit Windows 10.

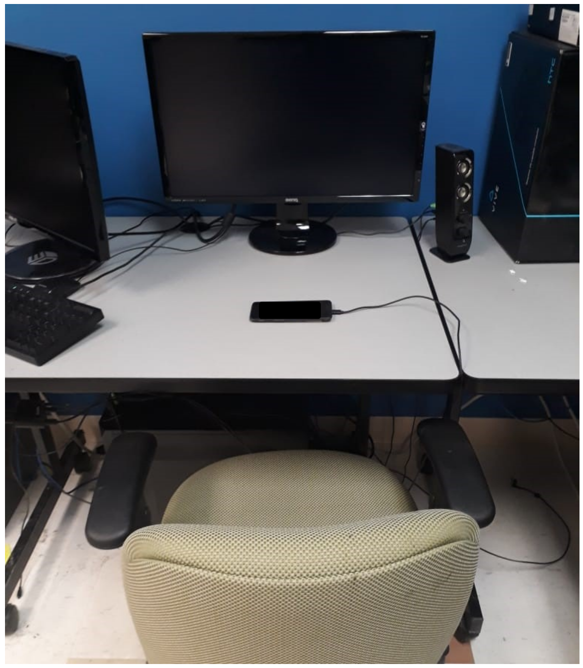

Figure 13: Apparatus showing relative position of desktop, smartphone, and participant chair.

The application started with the experimenter entering the experimental condition. After that point, text displayed in the middle of the screen said “Double tap to begin next trial”. The phone mirrored the desktop display using the Unity Remoting player which allowed the desktop application to receive touch input. In this fashion, participants were instructed to watch the desktop monitor and to only use the smartphone for interaction. The software detected the number and movement of fingers for making selections with a swipe gesture.

Upon beginning a trial, participants would see a series of menus and menu items they could interact with through swiping gestures on the phone. In each condition, the participant was shown a menu style with a target listed at the bottom of the screen. The user was tasked with selecting every instance of the target item from all the target menus.

The visual layout of each menu changed according to the menu style, the number of root menus and the number of items per menu. MenuCount had four rectangles along the bottom of the screen representing different menus which when selected showed a radial menu in the middle of the screen. ItemCount was made up of a larger radial menu with multiple items per swipe direction. Marking Menu used radial menus with two depths such that the second level of the menu was shown overtop the first level after the user made their first selection (Figure 14).

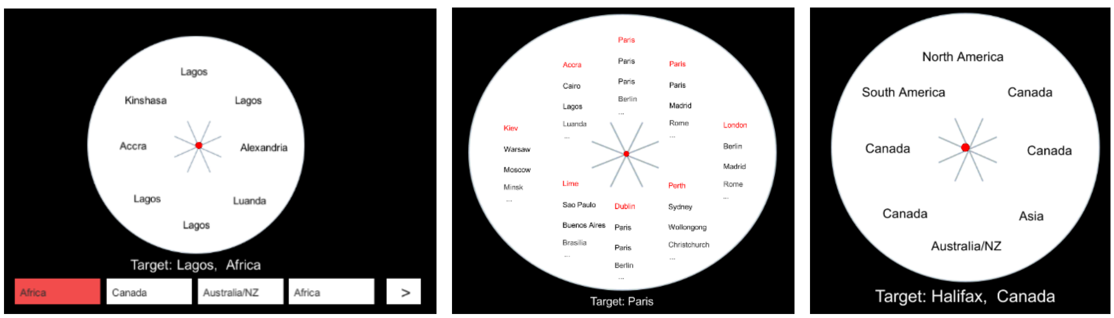

Figure 14: The 3 menu styles in their level 3, 8 x 8 configurations (from left to right) – a) MenuCount, b) ItemCount and c) Marking Menu

The software also recorded total trial time, number of correct selections and number of erroneous selections per trial. Execution time and number of fingers were recorded per selection.

4.3 Procedure

Upon arrival, participants were briefed on the design and purpose of the experiment. After providing informed consent, we gave participants instructions on how to complete the experimental task. Participants performed selections over 3 levels of difficulty using each of 3 different menu styles: MenuCount, ItemCount and Marking Menu. With each new menu style, participants were given a 3-minute training period where they were able to practice on an unrecorded trial. Difficulty level varied based on the number of menus, items per menu, and whether a mode change gesture was required. Trials started and ended when the participant double tapped the smartphone screen with two or more fingers. The first two trials of each block were practice trials. These had no target and were not recorded to give participants time to understand the task before it began. Most participants skipped the practice trials by double tapping with two fingers.

The task is taken from Bailly et al's study on Novice selection for wave menus [66] and required selecting a target city name from a series of menus that had the titles of countries and continents (see Figure 14). The menus were randomized such that half of them were the target menu. Similarly, the menu items (cities) were randomized and in each target menu, the target item appeared as half of the menu items plus or minus one. For example, with 8 menus and 8 items per menu, there would be 4 menus that included the target, and each of those menus would (randomly) contain either 3, 4 or 5 instances of the target item. The menu items (cities) always corresponded to the menu region. The total number of targets was the same for each trial in a level. This randomization ensured that participants could not count targets in each menu but would have to visually search for the targets to confirm none were left. After a target item was correctly selected it disappeared from the menu leaving a blank space (or dashed lines for ItemCount). When participants believed they had found all target instances, they double tapped with two fingers to end the trial. This task is used to best simulate visual search used in Novice selection without biasing the participant based on semantic knowledge. Bailly et al. [66] argue that Novice selection is guided more by semantics and as such when a user goes to find their target, they will be looking at all the related terms in the relevant menu and not just for the one item they select. This is the justification for selecting multiple targets within one menu. Compared to single target selection tasks, this reduces variation in results that are dependent on where participant begins their search. In single selection tasks, participants will find the target quicker if they happen to begin the search with that target. Selecting from variable numbers of targets requires a real search component regardless of where the search begins. Participants were instructed to complete each trial as quickly but accurately as possible.

Once a set of trials was completed, the experimenter set the participant up with the next difficulty level. Although participants could take breaks between trial levels, none opted to do so. After completing three levels of trials for each of the three menu styles participants were asked to complete a survey choosing their subjective preference for the menu styles.

4.4 Design

Our experiment employed a 3 x 3 within-subjects design. The independent variables included menu style (ItemCount, MenuCount and Marking Menu) and level of difficulty (levels 1, 2 and 3). Participants experienced each menu style in a different order that was counterbalanced according to a Latin square. For each menu style, participants progressed through 3 levels of difficulty.

Due to the different layouts of MenuCount and ItemCount, not all menu styles had the same number of menus and items per menu within a level of difficulty, however, they all had the same number of total items. The experiment conditions are summarized in Table 1, which also indicates the total number of menu items in each condition. For MenuCount and ItemCount, level 2 had the same total number of items as level 1 but required a mode-change gesture (Figure 15). Level 3 looked exactly like level 1, but with indicators that a mode change was required which doubled the number of items (Figure 14).

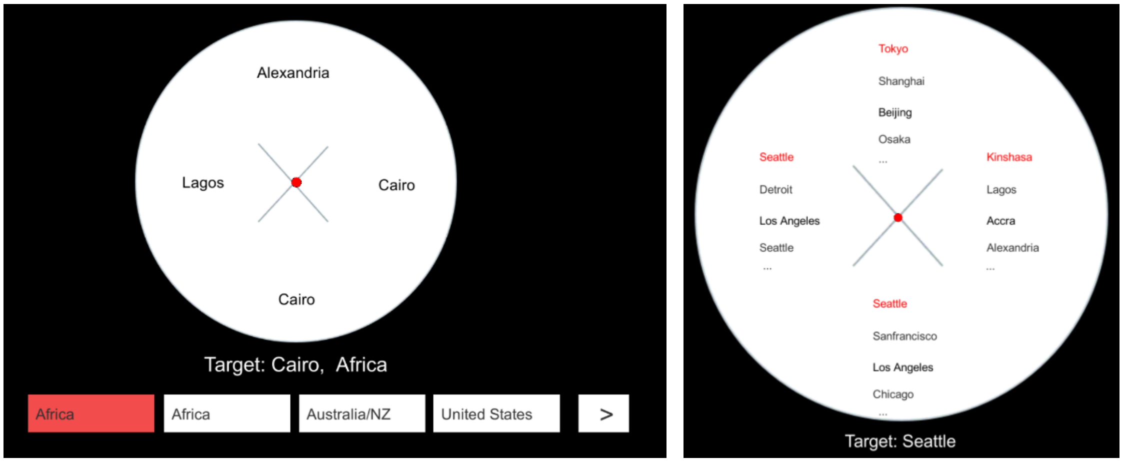

Figure 15: Menu layout for level 2 of MenuCount (left) and ItemCount (right)

	Menu style								
	Marking menu			MenuCount			ItemCount		
Difficulty Level	1	2	3	1	2	3	1	2	3
# Menu Items	4x8	8x4	8x8	4x8	8x4	8x8	8x4	4x8	8x8
Mode change required	No	Yes	Yes	No	Yes	Yes	No	Yes	Yes

Table 1: Conditions of the experiment conditions. Note that levels 1 and 2 of the CountMarks variants are not the same, but are ordered by whether or not a mode shift is required

Each difficulty level contained a block of 7 trials with each trial containing 7 (for difficulty levels 1 and 2) or 15 (for level 3) target selections, averaging 9.67 target selections per trial across the 3 levels of difficulty. Across 18 participants x 3 menu styles x 3 levels of difficulty x 7 trials each x 9.67 selections per trial, participants completed a total of 10965 target menu selections over 1134 trials. The experiment took participants approximately 1 hour to complete.

Dependent variables were *total selection time* (broken down into *reaction time* and *execution time*), *selection accuracy* and *search accuracy*. Total selection time is the trial time divided by the number of selections. Execution time is measured as the time from touching the screen to releasing the finger from the screen. This represents how long it takes to perform a selection gesture. Reaction time is measured as the total selection time minus execution time for selections in a trial. This represents the time it took for participants to understand the stimulus and begin a selection by placing their finger(s) on the touch surface. Selection accuracy is the number of correct selections divided by the total number of selections. Because participants could end the trial without selecting all the targets, search accuracy is the number of correct selections divided by the total number of targets presented.

Our main hypotheses were:

- H1: CountMarks will be a) faster and b) have greater accuracy (for both search and selection) than Marking menus.
- H2: Learning effects over the difficulty levels will result in faster and more accurate selections.
- H3: The more fingers used per selection, the greater the selection time and the lower the selection accuracy.

4.5 Results

We conducted a two-way (menu style x difficulty level) repeated measures ANOVA on each dependent variable. Note that Marking Menu requires two stroke gestures from the participant to make a selection (one to open the menu, one to select the target), however, due to a software error, only the final gesture was recorded. Because of this, the values for reaction and execution time below are doubled for Marking Menu selections. Doubling these values accurately reflects the total selection time for Marking Menu selections as recorded in each trial.

4.5.1 Total Selection Time

A summary of selection time broken down by reaction and execution time and by menu style and difficulty level is found in Figure 16.

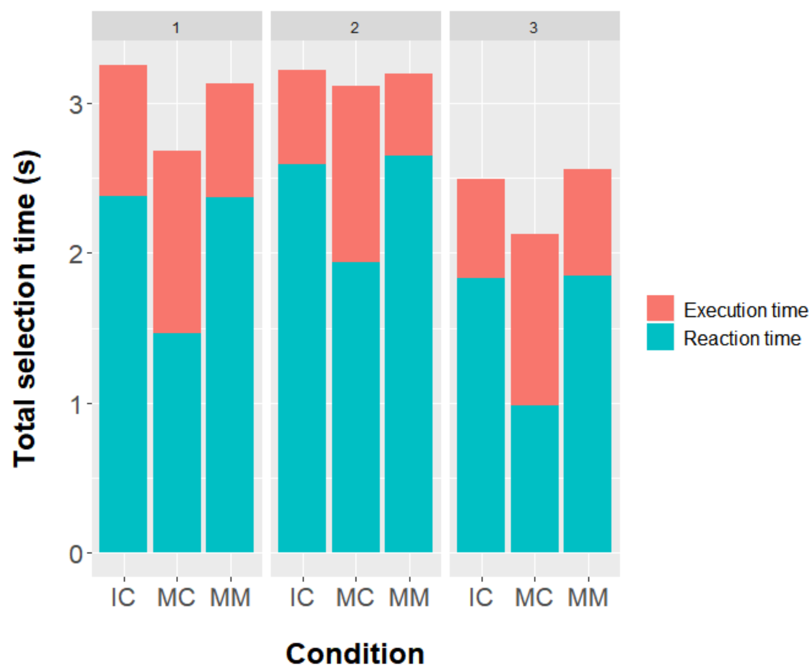

Figure 16: Mean Execution and Reaction time plotted against the menu style (IC = ItemCount, MC = MenuCount, MM = Marking Menu) and level of trials.

Total selection time is the time to complete each trial divided by the total number of selections in that trial and is the sum of reaction time and execution time for a selection. We found a significant main effect for menu style on selection time ($F_{2,34} = 25.03$,

$p < .001$). Post-hoc testing with the Tukey-Kramer test was performed and revealed that MenuCount offers faster selections (2.64s) than both ItemCount (3.00s) and Marking Menu (2.97s). This supports our H1a that the increased breadth of CountMarks allows for faster total selection time over menus with two depths.

ANOVA also found a significant effect for difficulty level on selection time ($F_{2,34} = 62.49$, $p < .001$). Post-hoc testing with the Tukey-Kramer test shows that difficulty level 3 offered faster selection (2.39s) than levels 2 (3.18s) and 1 (3.02s). This supports H2; since difficulty levels were always presented in the same order (1, 2, then 3), it corresponds to participants improving with practice. There was no significant difference between Levels 1 and 2, suggesting the added difficulty of introducing the mode change gesture offset any gains from learning effects. ANOVA did not detect an interaction effect between menu style and difficulty level.

4.5.2 Execution time

Execution time is the time from when the participant touches the screen until they release their finger for a selection. This reflects the actual time required to perform a selection, irrespective of any time mentally preparing for the task. Execution time was recorded for each selection and averaged out across the trial. The total execution time for Marking Menu is the sum of the execution times for the two strokes required to make a selection. Analysis of variance revealed a significant main effect for menu style on execution time ($F_{2,34} = 537.5$, $p < .001$). A post-hoc Tukey-Kramer test revealed both ItemCount (0.73s) and Marking menu (0.72s) offered significantly faster execution time than MenuCount (1.17s). Thus, this refutes our hypothesis H1a; CountMarks was not faster to execute than Marking menus. That MenuCount offers slower execution times is likely a reflection of where search takes place and will be discussed shortly. Analysis of variance also revealed a significant effect for difficulty level on execution time ($F_{2,34}$ = 33.51, p <0.001) that shows level 2 had quicker execution time (0.79s) than level 3 (0.87s), which in turn, had quicker execution time than level 1 (0.95s). This refutes our hypothesis H2. The mode change required in difficulty level 2 effectively splits the items from level 1 into two pages reducing the number of selectable directions from 8 to 4 (as seen in Figure 15). Selecting from fewer target directions likely allows participants to execute faster selections in level 2 than levels 1 or 3. Level 3 outperforming level 1 is likely due to a learning effect.

We conducted an analysis of variance on the number of fingers used for selection in both CountMarks variants against the execution time for each selection. We found a significant main effect for number of fingers used for selection on execution time ($F_{3,52}$ = 44.31, p < .001). A post-hoc Tukey-Kramer test revealed a significant effect where adding more fingers to a selection increases its execution time (Figure 17). This supports our hypothesis H3, that it is more difficult to perform selections with multiple fingers. There was no interaction effect found.

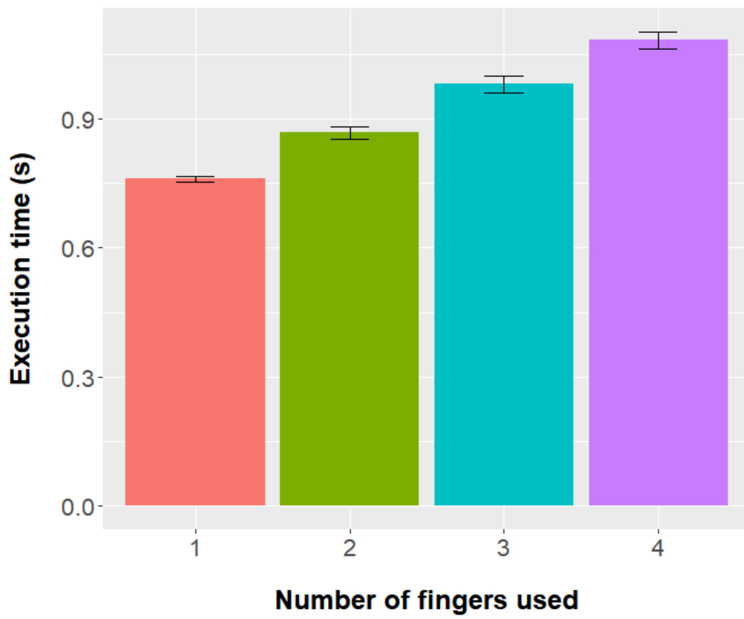

Figure 17: Execution time by the number of fingers used for selection

4.5.3 Reaction time

Reaction time is Total Selection Time – Execution Time for each trial. Reaction time is the time spent mentally preparing for a selection. This can include searching for the target and preparing the correct hand position to begin a selection. The total reaction time for Marking Menu is the sum of the reaction times for the two strokes required to make a selection. This requires time to first search for the correct menu, then for the target. An analysis of variance revealed a significant main effect of menu style on reaction time ($F_{2,34}$ =71.61, p <.001). A post-hoc Tukey-Kramer test indicated that selections using MenuCount offer quicker reaction time (1.46s) than ItemCount (2.26s) and Marking Menu (2.28s or 1.14s * 2 selections). This supports our hypothesis H1a that CountMarks will be faster than Marking Menus. We also found a significant main effect of difficulty level on reaction time ($F_{2,34}$ = 64.64, p <.001). A post-hoc Tukey-Kramer testing revealed that level 3 has a quicker reaction time (1.51s) than level 1 (2.07s) which has a quicker reaction time than level 2 (2.40s). This supports our H2 that participants should get faster with practice. Like total selection time, we speculate that Level 2 being slower than Level 1 reflects the difficulty added by introducing the mode change double tap gesture. There was no interaction effect found.

4.5.4 Selection Accuracy

Selection accuracy is the number of correct selections for a trial divided by the total number of selections made in that trial. A summary of selection accuracy broken down by menu style and difficulty level is found in Figure 18.

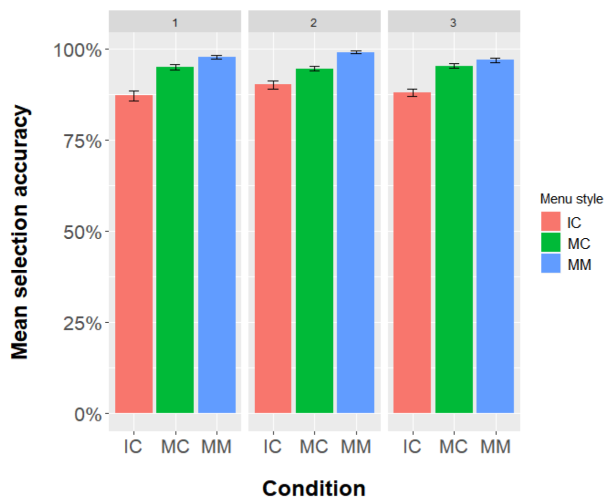

Figure 18: Mean selection accuracy by menu style and difficulty level

Analysis of variance revealed a significant main effect of menu style on selection accuracy ($F_{2,34} = 14.26$, $p < .001$). A Tukey Kramer posthoc test revealed that Marking Menu was significantly more accurate (97.91%) than MenuCount (95.00%) which is more accurate than ItemCount (88.44%). This does not support our H1b. By introducing the added complexity of extra fingers to perform selections, CountMarks has more opportunity for errors to be made and thus has lower selection accuracy than Marking Menu. The particularly low selection accuracy of ItemCount will be discussed further below. There was no significant effect of difficulty level on selection accuracy. This does not support our H2 and could be the result of participants being focused more on making the correct selection while learning the different menu styles, and less on performing the selections quickly. There was no interaction effect found.

4.5.5 Search Accuracy

Search accuracy is the total number of correct selections divided by the total number of targets for that trial. This represents the likelihood of a participant accurately selecting their intended target. A summary of search accuracy broken down by menu style and difficulty level is found in Figure 19.

We found a significant main effect of menu style on search accuracy ($F_{2,34} = 100.98$, $p < .001$). A post-hoc Tukey-Kramer test showed that MenuCount (97.43%) and ItemCount (96.45%) are significantly more accurate in finding targets than Marking Menu (92.16%). This supports our H1b that CountMarks can improve search accuracy by increasing menu breadth. We also found a significant main effect for difficulty level on search accuracy ($F_{2,34} = 3.73$, $p < .05$), a post-hoc Tukey test was performed that shows level 2 to be significantly less accurate than level 1 or 3. This does not support our H2 and is likely due to participants missing items behind the newly implemented mode-change gesture. There was no interaction effect found.

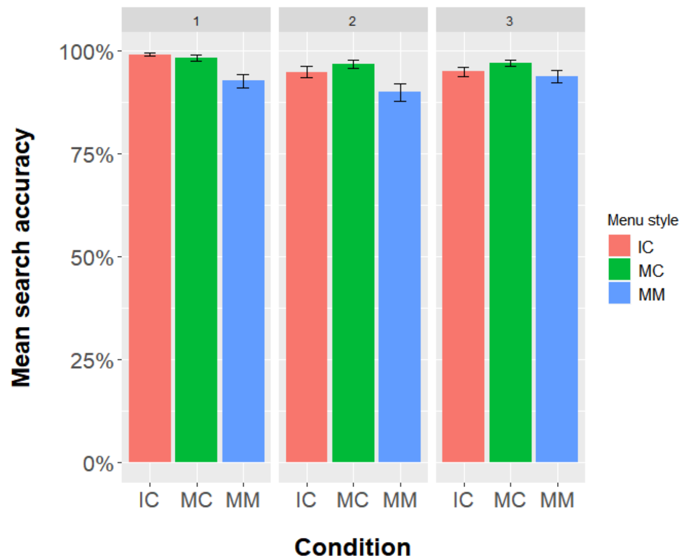

Figure 19: Mean search accuracy by menu style and difficulty level

4.5.6 Survey

Participants were asked to identify which menu style they thought was the easiest to learn, which was their favorite, which was the quickest to perform and which was the most tiring to perform. For each question, the participant could select only one menu style. Marking Menu was ranked as the easiest to learn (56% of responses), quickest to perform (44%) and the least tiring to perform (22%). MenuCount was ranked most as the favorite menu style to use (44%). The full breakdown is shown in Figure 20. We note that during the 3-minute training task none of the participants showed difficulty understanding either variation of CountMarks.

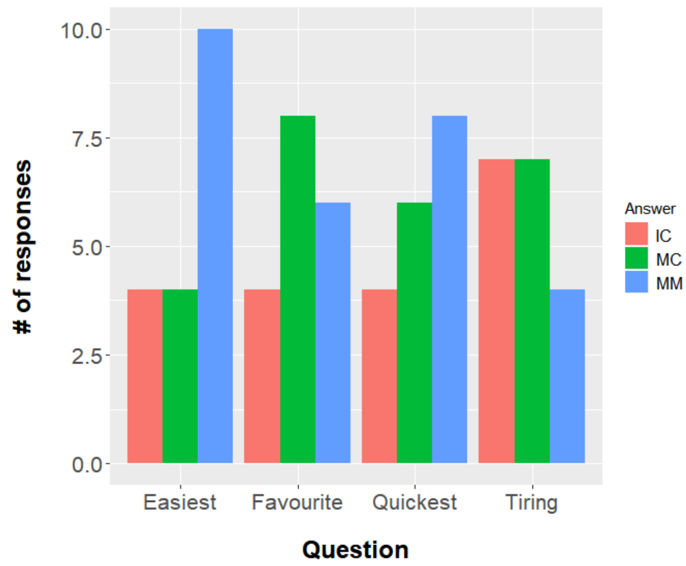

Figure 20: Survey response of favorite interaction technique by factor

4.6 Discussion

CountMarks is designed to reduce or eliminate depth in Marking menus by providing more accessible menu breadth using multiple fingers for selection. The goal of our experiment was to compare two variations of CountMarks against traditional Marking menus. Our results are summarized below:

4.6.1 CountMarks vs Marking menus

This study finds that the MenuCount variant of CountMarks outperformed Marking Menu in selection speed and search accuracy while Marking Menu performed better in selection accuracy. Despite the strong performance from MenuCount, Marking Menu was ranked as the easiest to learn and the least tiring to perform by participants. This is expected as Marking Menu only required one finger to use, which may be a more familiar interaction for participants. Interestingly, Marking Menu was also ranked subjectively as the quickest menu style for performing selections despite the results showing otherwise. We believe this may be due to its more simplistic nature. Two participants did mention that they believed MenuCount would be the quickest with more practice. What was not expected is that despite favorable rankings for Marking Menu, MenuCount was narrowly ranked most as participant's favorite menu style. Whereas Marking Menu required repetitive swiping with one finger, perhaps participants found MenuCount more engaging because of the added complexity of choosing how many fingers to use for each selection.

Overall, our results align with Lepinski et al's study [21] comparing multitouch marking menus to the multi-stroke marking menus (the Marking Menu style in this study). Lepinski et al found that by increasing menu breadth with multi-finger chording gestures, that their multi-touch marking menu could decrease selection speed. While Lepinski et al did not find any significant difference in selection accuracy between their technique and multi-stroke marking menus, we believe this may be due to their smaller sample size (12 participants) and not reflective of worse performance for CountMarks.

4.6.2 CountMarks variants

Of the two CountMarks variants, MenuCount outperformed ItemCount on almost every performance measure. MenuCount was faster, more accurate and was ranked more often as a participant's favorite menu style. Of particular note is how much worse ItemCount's selection accuracy was. We believe this may be due to two factors: item location did not always perfectly match its selection direction and participants may have been confused how many fingers to select with due to blank spaces left by previously selected items. Because ItemCount places a very large number of items on the screen at once, it was often not clear which direction the target item was in. For example, if the target was at the bottom of the bottom left menu (e.g. Seattle in Figure 21), it may look like the participant should swipe to the bottom of the screen to select it, instead of swiping to the bottom left. Alternatively, participants may have been confused by how many fingers were required for selection. ItemCount had over twice as many errors from selecting an item that had already been selected compared to MenuCount. Whenever a target was correctly selected it was removed and replaced with dashed lines to signify empty space. Perhaps participants believed that after a menu item was removed, the remaining items were supposed to move to take that item's place. For example, if as in Figure 21, the first item in a linear submenu was correctly selected (e.g. Seattle in the rightmost menu), the participant may then believe that the menu items move up in the menu such that the 2^{nd} item now needs only 1 finger to select it. As a result, if the participant tries selecting the 2^{nd} item they may only use 1 finger to do so which would result as an error. In fact, all selections that used fewer than 4 fingers for ItemCount selection recorded twice as many errors from these empty selections (if the fourth item is selected, no item would take its place).

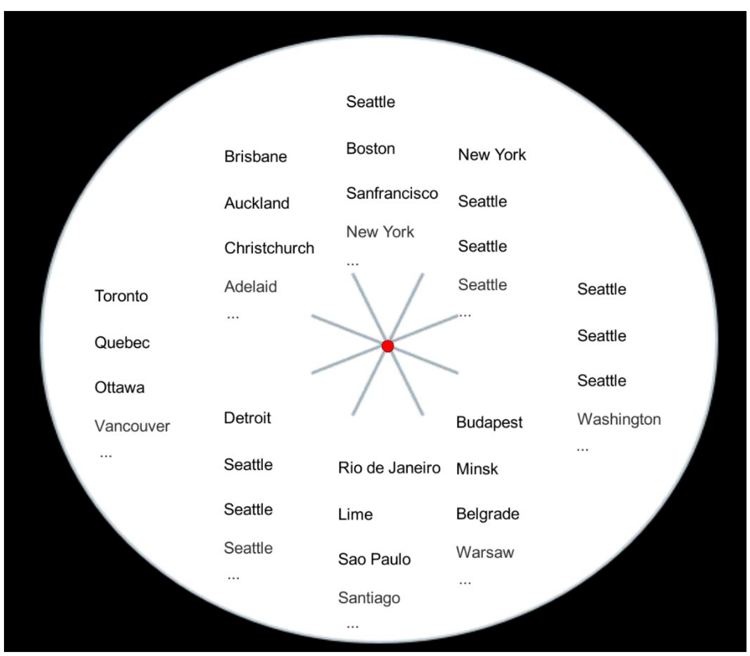

Figure 21: Example of how ItemCount arranges menu items for selection

The two CountMarks variants also had vastly different execution and reaction times. ItemCount requires the user to search for the target amidst many options, place the correct number of fingers on the screen and then swipe in the direction of the target item. This long search component happens before the execution begins, so the time required for search is added into its reaction time. In contrast, MenuCount requires the user to first search to find the target menu from a small number of options, and only once they have placed down the correct number of fingers do they search for the target item. Because of this, MenuCount had the longest execution time out of the three menu styles. Despite the longer execution time MenuCount still had the quickest total selection time, showing that by breaking the search component up into smaller, more easily accessible and reversible components (menus can be quickly previewed by placing a number of fingers down and releasing them) it was quicker to complete a selection. ItemCount did have better search accuracy than MenuCount. We believe this reflects the ability for participants to see more items at once and determine whether any targets remain. The obvious drawback of this is that search takes longer with so many items and selection time suffers.

4.7 Limitations

The findings for total selection time are limited by how it was recorded. By recording selection time as total trial time divided by the number of selections, the time to end the trial and search through remaining menus is included. While this does reflect the time required for users to fully complete their search, it should otherwise not be included in the individual selection times. Similarly, this poses a limitation on reaction time because instead of calculating a reaction time per selection, it was averaged out by trial based on the total selection time described above.

Secondly, the results of execution time for Marking Menu is limited by recording only the 2nd of two strokes in a selection. Because of this we doubled the execution time of the single strokes instead of using the sum of two individual strokes as the total execution time. It is unclear the effect this has on the results, but it only affects the reaction and execution times, not the total selection times as these were recorded separately. Regardless, the results shown are consistent with the length of time it took participants to complete each trial and so the overall finding of MenuCount outperforming both ItemCount and Marking menus in terms of selection time and search accuracy remain valid.

Finally, there are concerns of ecological validity with both the chosen task and our participant sample. The task is not representative of real-world use and our participants, being primarily young students, may not be representative of the general population. The next chapter will explore the performance of the MenuCount variation of CountMarks in various device positions and user stances.

Chapter 5: User Study 2

We conducted a second user study to evaluate MenuCount in more realistic use cases. Consistent with past studies on marking menus [66], we tested participants' performance in both a *Novice* task that required searching for targets prior to selecting them and an *Expert* task which required no search. Participants completed these tasks with the smartphone mounted on the hand, and the leg while standing or walking. The ethics protocol for this project was reviewed by the Carleton University Research Ethics Board with CUREB-B clearance.

5.1 Participants

We recruited 20 participants, average age 24.6 years old (*SD* = 4.31 years), 60% were female, all were right-handed. Average height was 170.5cm (SD = 9.69cm). Nine participants reported often texting while walking, 8 sometimes texted while walking and 3 others always, rarely, or never did so. Six participants had participated in the previous CountMarks study (see Chapter 4:). Participants were recruited via posters around campus as well as through a post on a Carleton research Facebook page.

5.2 Apparatus

We used a Samsung Galaxy S8 smartphone running Android 9.0 as the touch input device. We used a Microsoft HoloLens as the HMD, to more realistically simulate usage scenarios with smart glasses. The software was built in Unity and used Unity’s built-in networking tools, UNet, to take input from the Android app to a desktop computer. The desktop display was wirelessly mirrored to the HoloLens using the HoloLens Remoting app.

The desktop computer had an Intel Core i7-2600K CPU at 3.40GHz with 16 GB of RAM, running 64-bit Windows 10. Selection was performed on the smartphone and using UNet's native scripts, the software sent updated cursor position and menu information to the HoloLens every 0.05s. Depending on the condition, the smartphone was either held in the participant's non-dominant hand or mounted with Velcro on the upper part of the leg, so that their dominant hand could comfortably reach it. Participants walked on a Tempo Fitness 610T treadmill. Figure 22 depicts the hardware setup.

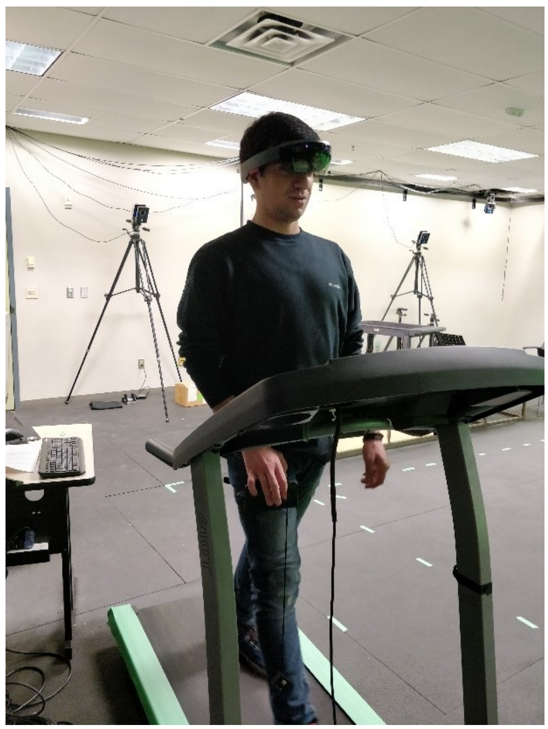

Figure 22: A pilot study participant performing CountMarks on a smartphone mounted to his leg to interact with the Microsoft HoloLens while walking

For the Novice task, the software recorded the reaction time, execution time, and selection and search accuracies. The expert task did not track search accuracy but did track the types of errors made and the end positions of each selection in addition to the other measurements.

5.3 Procedure

5.3.1 Setup

Upon arrival, participants were briefed on the design and purpose of the experiment. After providing informed consent, participants were given a demonstration of how to properly position the HoloLens to see the full display. We began the experiment by identifying which way participants thought was "up" when swiping on their leg. This is non-obvious because "up" depends on the participant's frame of reference, and could refer to the top of the participant, the forward direction they are facing, or the direction their hand is pointing at the time of interaction. Participants were asked to stand about 3 feet in front of the computer monitor and were shown an image of an arrow pointing to the top of the monitor. They were asked to perform a swipe gesture in the direction of the arrow at the point where their hand rests at their leg. The experimenter held the input device to their leg for this selection and recorded the direction of their gesture.

Participants then completed a demographic questionnaire before receiving instructions on how to use CountMarks to complete the Novice Task. They were given 3 minutes to practice CountMarks with an unrecorded Novice trial. During the practice trial, participants were seated wearing the HoloLens with the input device horizontally on the desk. Participants would receive an additional 3 minutes of practice time for the Expert task after completing Novice trials.

After practicing CountMarks, we determined a comfortable walking speed for each participant to use in the remainder of the experiment. We asked participants to walk on the treadmill at a speed of 2.0 mph while wearing the HoloLens. Participants could increase or decrease the speed by up to 0.5 mph (the smallest increment on the treadmill).

No participant chose to go higher than 2.0mph, but 12 participants chose to go slower. After determining walking speed, participants stepped off the treadmill, and we attached Velcro straps to their leg to mount the smartphone. The straps were positioned where the fingers naturally rest by the leg when the participant relaxed their arms.

5.3.2 Task

Participants performed both Novice and Expert selection trials in each of four conditions consisting of the combinations of standing vs walking and holding the smartphone vs mounting the smartphone on the leg. With Novice selection, the user must visually search for their target. Expert selection occurs when the user already knows the gesture to select a given target, and thus just execute the selection. All participants started with Novice trials before the Expert trials. In pilot testing, we found that Novice trials take substantially longer to complete than the Expert trials. Consequently, participants completed fewer Novice trials than Expert trials, so each task took around the same amount of time. Participants performed 4 trials per condition for the Novice task and 8 trials per condition for the Expert task. The order of the conditions was counterbalanced and was kept the same for Novice and Expert tasks.

A trial started when the participant performed a double tap gesture with 3 or more fingers. In walking conditions, the participant stood on the treadmill while it moved at a comfortable speed determined earlier. In the standing conditions, participants stood on the deactivated treadmill. The Novice task was identical to that used in Study 1 (see Chapter 4.4).

The Expert task simulated expert usage where the participant had memorized item locations and did not require visual search. Instead of showing a menu, it instead showed a number and a direction arrow. This indicated which direction they should swipe, and with how many fingers, simulating true expert usage (Figure 23).

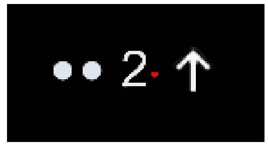

Figure 23: An example of selection to be made in the Expert task require a double tap with one finger followed by a two-finger swipe gesture up.

Some selections required a double tap before the swipe gesture. Such trials were indicated by two dots before the number. This simulated the item being accessible only after performing a mode change, similar to some selections in the Novice task. For each trial in the Expert task, there were 4 groups of 3-5 selections. Each grouping combined a different number of fingers and some included double tap dots. Within each grouping, all swipe directions were different. This emulated the Novice task since each Novice trial had targets on 4 menus with each menu containing 4 targets, plus or minus 1. After the participant performed a selection, they received feedback indicating if their selection was successful (a green checkmark or red x), and the target would change. Participants were instructed to complete each trial as quickly but accurately as possible. Both the Novice and Expert tasks had a tiny red dot cursor in the middle of the screen to give feedback as to which direction their fingers were moving during selection. Participants were given 5-minute breaks between the conditions where they could remove the HoloLens. After completing all 4 conditions for both tasks, participants completed a questionnaire about their experience with the different conditions.

5.4 Design

The experiment employed a 2×2×2 within-subjects design, with the following independent variables and levels:

- Device position (hand, leg)
- User stance (standing, walking)
- Task (Novice, Expert)

The Novice task was always performed first. The order of device position and user stance conditions were counterbalanced via a Latin square.

The dependent variables were *reaction time*, *execution time* and *selection accuracy*. Reaction time is measured as the time from the beginning of the trial or end of the last selection to the first touch of the smartphone screen to start a selection. This represents the time it took for participants to understand the target stimulus before beginning their selection. Execution time is measured as the time from touching the screen to completing the selection. Selection accuracy is the number of correct selections divided by the total number of selections. The Novice task includes *search accuracy* as a dependent variable. and is measured by the number of correct selections divided by the number of targets. This does not apply to the Expert task because no search is required. The Expert task tracked the final XY coordinates of the red cursor as well as the selection itself to categorize error types.

In total, 20 participants were tested across 2 user stances with 2 device positions through 2 tasks. In the Novice task, participants completed 4 trials consisting of 15 selections each, for a total of 20 participants x 2 user stances x 2 device positions x 15 selections per trial * 4 trials = 4800 Novice selections across 320. Participants completed twice as many trials for the Expert conditions, or 9600 Expert selections across 640 trials. Our main hypotheses were:

- H1: Both the leg position and the walking stance will have slower selection (execution and reaction) time.
- H2: Both the leg position and walking stance will have lower selection accuracy
- H3: The Expert task will have greater selection (execution and reaction) time and lower accuracy than the Novice task.

5.5 Results

We analyze our results using two-way repeated measures ANOVA for each dependent variable. The results of participants who also participated in Study 1 did not significantly differ from the rest of the sample in this study.

5.5.1 Reaction time

A summary of the results for reaction time broken down by task and condition is found in Figure 24.

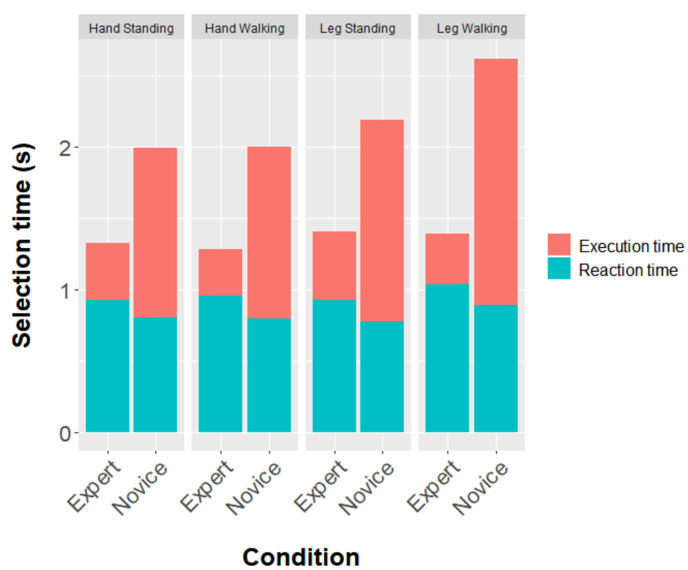

Figure 24: Execution and reaction times broken down by task and condition

Analysis of variance revealed a significant main effect of task on reaction time ($F_{1,19}$ = 83.73, p<.001), where participants showed quicker reaction times with Novice selections (0.820s) than with Expert selections (0.960s). This does not support H3 and may reflect the Novice user's desire to start the selection so they can search for their target, whereas the Expert user may need to prepare the correct number of fingers and correct selection direction.

Analysis of variance revealed a significant main effect of device position on reaction time ($F_{1,19}$ = 5.68, p<.05). Holding the device in the hand offered quicker reaction time (0.897s) than interacting with the device on the leg (0.933s). This supports H1 and is likely because the leg is an unfamiliar place for interaction it may take more thought to begin a gesture there. Analysis of variance revealed a significant main effect of user stance on reaction time ($F_{1,19}$ = 16.74, p<.001), where participants showed quicker reaction times while standing (0.884s) than while walking (0.944s). This supports H1 as walking adds complexity to the task.

Interaction effects were found between device position and user stance ($F_{3,57}$ = 9.86, p <.01). See Figure 25. When walking, reaction time was much higher when the smartphone was mounted on the leg than with the device in hand. In contrast, device position had little impact when the user was standing.

Figure 25: Interaction effect between user stance and device position on reaction time

5.5.2 Execution time

A summary of the results for execution time broken down by task and condition is seen in Figure 24.

Analysis of variance revealed a significant main effect of task on execution time ($F_{1,19}$ = 1644.20, p<.001), where participants executed Expert selections quicker (0.390s) than Novice selections (1.382s). This supports H3 as Novice selections require the participant to search for the target after placing their fingers down and beginning to execute the selection. In contrast, Expert selections are executed knowing which direction to swipe in.

Analysis of variance revealed a significant main effect of device position on execution time ($F_{1,19}$ = 50.39, p<0.001), where participants perform selections quicker with the device in the hand (0.637s) than with the device on the leg (0.787s). Similar to the results of reaction time, this supports H1 as the leg is an unfamiliar place for interaction to occur and may take more thought to execute a gesture there. Furthermore, we anticipated participants may perceive "up" differently when swiping on their leg. This difference in perception may cause participants to correct their thinking when executing selections with the device on their leg and thus increase the execution time.

No significant main effect was found for user stance on execution time. This does not support H1 as we would expect walking to add more difficulty to each selection and therefore increase execution time for the walking stance.

Interaction effects were found between device position and task ($F_{3,57} = 39.03$, $p <$ .001), and between user stance and task ($F_{3,57} = 22.25$, $p < .001$). See Figure 26. Novice selections were more strongly affected more by changes in both user stance and device position.

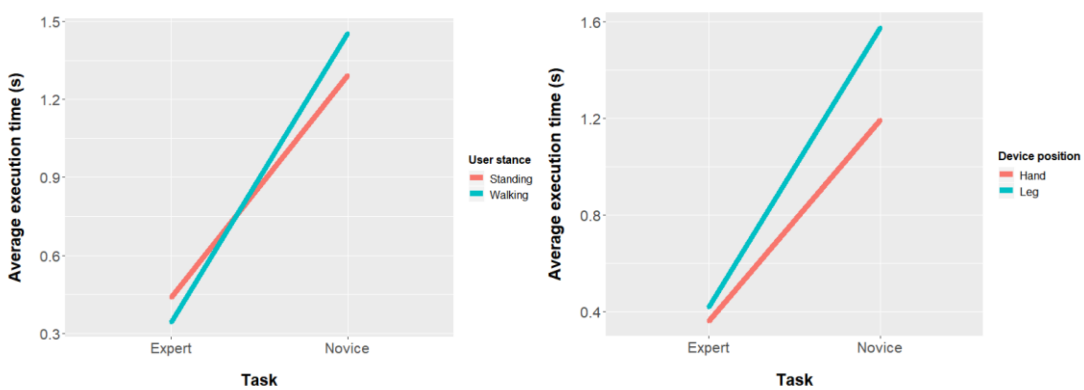

Figure 26: Interaction effects for task vs user stance (left) and task vs device position (right) on execution time.

5.5.3 Selection accuracy

A summary of error percentages broken down by task, condition and number of recorded fingers is seen in Figure 27.

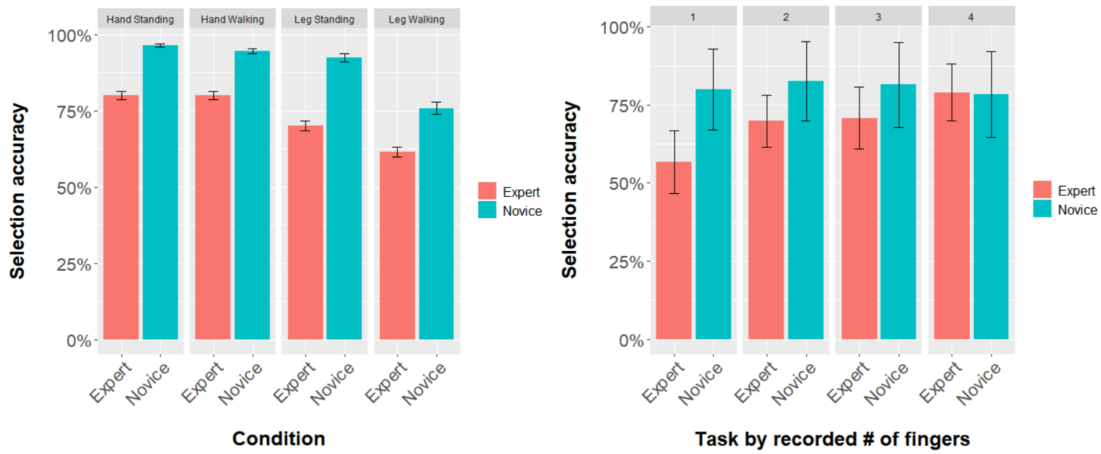

Figure 27: Selection accuracy by condition vs task (left) and by recorded number of fingers vs task (right). Error bars showing SE.

Analysis of variance revealed a significant main effect of task on selection accuracy ($F_{1,19}$ = 303.85, p<.001), where participants were more accurate making Novice selections (90.21%) than Expert selections (73.11%). This supports H3 and is likely because the Expert task gave no feedback for whether the participant was using the correct number of fingers and it did not display boundaries of the swipe directions. While expected, the particularly low accuracy for Expert selections was surprising and will be discussed below.

Analysis of variance revealed a significant main effect of device position on accuracy ($F_{1,19}$ = 201.98, p<.001), where selections with the device in the hand are more accurate (85.46%) than selections with the device on the leg (72.13%). As before, this is likely because the leg is an unfamiliar interaction location.

Analysis of variance revealed a significant main effect of user stance on selection accuracy ($F_{1,19}$ = 40.10, p<0.001), where participants were more accurate while standing

(81.94%) than while walking (75.94%). This supports H2 as walking adds complexity to the task.

Interaction effects were found between device position and user stance ($F_{3,57} = 7.97$, p <.01), and between user stance and task ($F_{3,57} = 10.01$, p <.01). This is shown in Figure 28 where user stance affects accuracy more in leg-mounted conditions than hand-held conditions and affects Novice accuracy more than Expert accuracy.

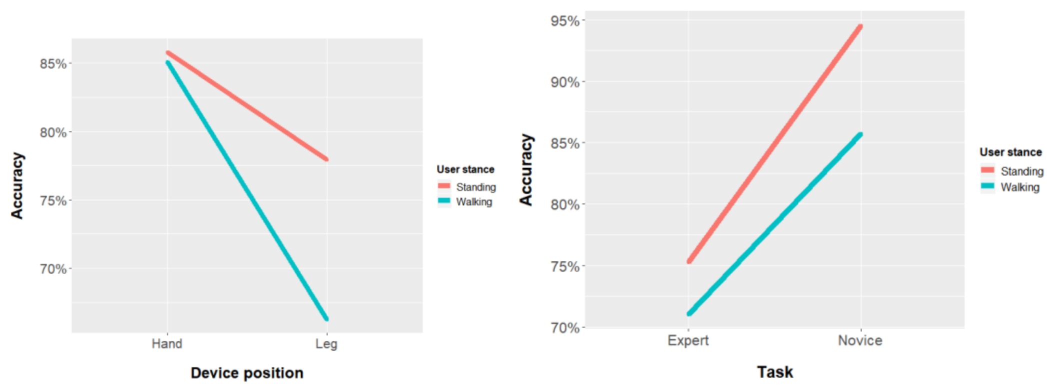

Figure 28: Interaction effects of device position vs user stance (left) and task vs user stance (right) on selection accuracy.

A further breakdown of errors for Expert selections is found in Figure 29. Because the Expert task had a specific target for each selection, we could identify the types of mistakes participants made. These will be discussed below.

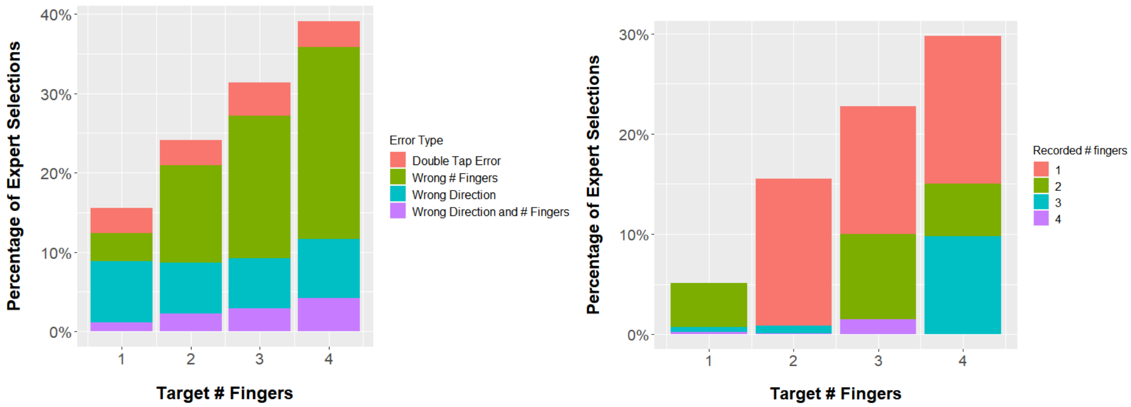

Figure 29: Expert selection error rates by number of fingers and error type (left) and expert selection error rates using the wrong number of fingers by target number of fingers vs detected number of fingers (left).

For Novice selections, no significant main effects were found for user stance or device positioning on search accuracy. We note that the search accuracy for novice selection in this study was lower (93.55%) than in our first study (97.43%).

5.5.4 Data correction and limitations

Due to a software error the final trial of some of the first 5 participants was not recorded. Of the remaining 13649 recorded selections, we filtered selections faster than 0.001s reaction or execution times, and those with longer than 10 s reaction or execution, leaving 13337 selections. These times were either too fast or slow to be intentional and were thus software or hardware errors. Additionally, some participants noted that the double tap stopped working during Expert trials. These participants were instructed to finish the task to the best of their abilities. Figure 29 shows these types of errors, along with naturally occurring Double Tap errors account for less than 5% of all selections. Finally, because the input device had to connect wirelessly to a desktop computer which then was wirelessly tethered to the HoloLens, there could be noticeable lag at times. This lag was only visual as the actual selection occurred on the input device.

5.5.5 Calibration, Hand Positions and Survey results

At the beginning of the study we asked participants how they would swipe “up” on their leg using an arrow pointing to the top of the computer monitor as a reference. Results showed that 40% said they would swipe from back to front, 30% said they would swipe towards their feet and 30% said they would swipe towards their head. We were surprised by the even variety of responses and suggest that future work explore how this preconceived orientation of swipe gesture affects performance.

In the hand-held device position condition, we observed participants’ hand positions while they held the smartphone. Because multi-finger marking menu selection is not a usual task to perform on mobile phones, participants were permitted to hold the phone however was comfortable. Participants held the phone in various grips, see Figure 30.

Figure 30: Phone grips across participants in the hand-held conditions. From left to right (a) 20% gripped the phone on fingers between pinky and thumb, (b) 20% on their fingers held out, (c) 20% on their hand sideways between thumb and index finger (d) 15% on their fingers against the palm in a cusped hand, and (e) 10% on the tips of their fingers away from palm,

The variety of unprompted grips should be considered whenever asking users to perform marking menu gestures on a mobile device. We also noted that all but one participant held the phone around chest height.

Finally, participants completed a 5-point Likert-scale questionnaire asking how comfortable and easy each condition was. Results are summarized in Figure 31, showing that participants' feelings of ease and comfort of performance are largely tied to the efficacy of those interactions. We believe this may be a response to our input device being a smartphone and that participants rated their feelings based on this bias of what interacting with a smartphone should feel like. We suggest the comfort and ease of performing marking menu selection in mobile conditions be further explored with this bias removed.

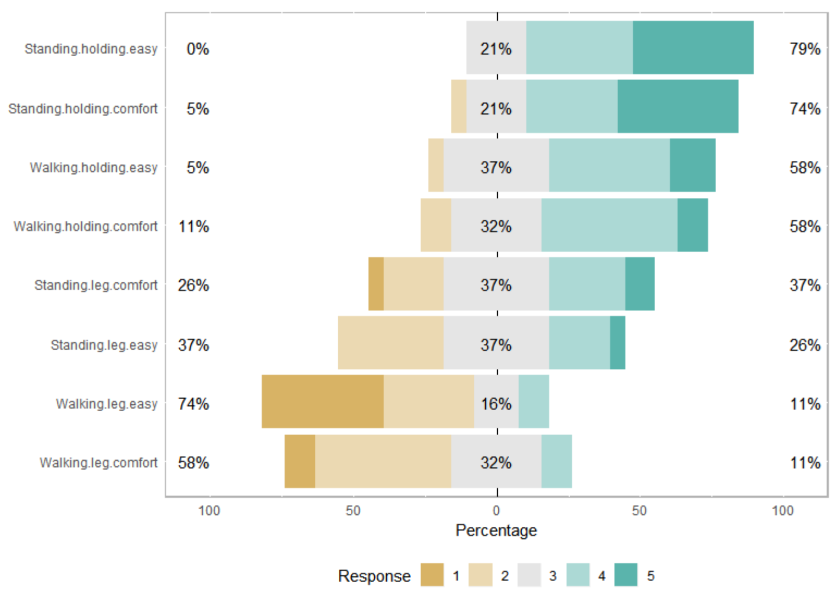

Figure 31: Survey response data, 1 is low 5 is high. Questions were asked about the ease and comfort of performing CountMarks in different user stances and at different device positions.

5.6 Discussion

The goal of this study was to assess realistic use cases of CountMarks by testing different device positions (the hands vs the leg) with different user stances (standing and walking) and analyzing the effects for both in Novice and Expert selection tasks. Our results are summarized below:

5.6.1 Novice vs Expert Selection

One of the main benefits of marking menus is their ability to facilitate the transition from Novice to Expert selection [7]. Novice selection requires the user to search for the target. With Expert selection, they simply perform the correct gesture. We will discuss the results for selection speed and accuracy below.

5.6.1.1 Selection speed

Despite search being required for Novice selections, reaction times differed between the two tasks by only 0.14 seconds, whereas execution time differed by almost a full 1.00s. CountMarks requires two search components: the first to choose a menu, and the second to find the target within the menu. The smaller than expected difference is likely because participants returned to the same menu multiple times already knowing where to find the target in the Novice task. So, while the first selection would require reaction time to search for the correct menu, subsequent reactions may be performed more like Experts. Additionally, Expert reaction time may be greater than expected if the participant needs to plan both the number of fingers to use and the direction to swipe in before placing their fingers down, instead of just the number of fingers needed to open a menu in a Novice selection.

The bigger difference between Novice and Expert is seen in execution time where participants were over 3.5 times quicker to execute Expert selections. This reflects how Novice users wait for the menu to open before searching for the target item. Table 2 shows that for CountMarks, Novice selection speed (when performed standing with the phone in the hand) outperforms (or equaled) Expert selection from most other marking menus. Our Novice selections were also more accurate. While our Expert selection was substantially faster than our Novice selection, it suffered greatly from poor selection accuracy which will be discussed below.

Name of technique	**Selection type**	**Selection speed**	**Selection accuracy**
Kurtenbach marking menu [16]	Expert	~2.3s	80%
Multi-stroke marking menu [16]	Expert	~2.3s	93%
Polygon menu* [19]	Expert	~2.0s	95%
Multi-touch marking menu [21]	Novice	2.37s	93.6%
Multi-touch marking menu [21]	Expert	0.81s	85.1%
CountMarks **	Novice	1.99s	96.5%
CountMarks **	Expert	1.33s	80.2%

Table 2: A comparison of performance for different marking menu styles using 8 x 8 menus. Note that while the methodologies are not identical for each evaluation, they correspond to the same or similar tasks. *Polygon uses 12 x 12. **CountMarks in the standing, handheld condition.

We also found that double tap gestures added only 0.87s to an Expert selection, showing that they are quicker to perform than a full Expert selection (whose average selection time without a double tap 1.50s). Therefore, if a target is not found in the first menu level it is quicker for the user to change modes to find more items than to make a full selection and access a deeper menu level. This validates our design decision to use a mode changing gesture rather than adding menu depth with an additional selection.

5.6.1.2 Accuracy and Error Rates

In his evaluation of marking menus, Kurtenbach [7] noted that acceptable accuracy depends on the consequence of errors and how difficult it is to undo errors. Compared to other styles of marking menus (Table 2), ideal accuracy should be around 95%, but these studies only tested selection while sitting at a desktop computer with a stationary input device. It is unclear what an acceptable error rate is for mobile interactions, especially when standing or walking.

In our study, the largest factor affecting accuracy was the task, where Expert selections were 16.91% less accurate than Novice selections. We attribute two main reasons for this: The Expert task gives no feedback for how many fingers the user is selecting with, and we believe there were software errors in detecting the number of fingers used for a selection. The Novice task provides visual feedback in the form of menus appearing, corresponding to the number of fingers the user is selecting with. This helps ensure that users perform selections with the right number of fingers. In contrast, the Expert task gives no such feedback; the user may accidentally use the wrong number of fingers without realizing it or may not be aware if the device is not sensing the intended number of fingers. We argue that the exceptionally low accuracy rate in the Expert task is primarily due to software errors of finger recognition rather than user errors. Additionally, a software bug sometimes prevented double taps yielding some additional errors. Figure 29 shows an unequal distribution of error types where almost

30% of 4-finger selections had errors due to the wrong number of fingers being used by the participant. In contrast, this only occurred with 5% of 1-finger selections. Almost 15% of selections requiring 4-fingers, recorded the participant using 1 finger. If users were simply had difficulty using the correct number of fingers, then we would expect a more even distribution of errors based on number of fingers used. While there is the possibility that participants get confused and use the wrong number of fingers when no feedback is presented, we rarely recorded instances where the participant uses 4 fingers to select a 1 finger target.

Lepinski et al. [21] created a similar style of multi-touch marking menu and found a related problem. Their Expert selections had similarly low accuracy (Table 2) which they attributed to contact points being lost when they were moving too quickly. We tried to address a similar issue by not allowing the recognized finger count to change while contact points were moving. However, if participants began Expert selections with their fingers in motion (since they begin the selection knowing which direction to swipe in) this may cause the software to record only the first few fingers that touch the screen. This is not a problem in the Novice task because participants usually wait for visual feedback of the correct menu appearing before moving their fingers to make a selection. This is speculation, but it mirrors the reactions of participants during the study. Most participants mentioned some sort of frustration or surprise when starting the Expert task because they were making wrong selections despite feeling they were making the correct ones. In the post-study questionnaire, 3 participants specifically mentioned they thought the device had finger recognition issues with 3 others referring only to poor recognition on 4 finger swipe gestures. Furthermore, this explanation is coherent with the findings from both our

Novice task and Lepinski et al's Novice task where participants showed no difficulty articulating the correct number of fingers for a selection.

A pattern also emerges from errors caused by performing selections in the incorrect direction. These errors contributed to less than 10% of all Expert selections. We believe most of these errors have two main causes: the participant has a different understanding of which way "up" is when swiping on the device, and that participants do not swipe in straight lines. At the beginning of our experiment, we asked participants to swipe in the direction of an up arrow on their leg. Participants swiped in one of three directions: towards their head, towards the front of their body or towards the floor. We design CountMarks so that swipe orientation is fixed regardless of if the smartphone is hand-held or attached to the leg. Consequently, leg interactions are inverted: the phone goes from being upright when facing the user in their hand to upside down when facing away from the user on the leg. We argue that such a design make sense if CountMarks is performed when transitioning from performing gestures on the top of the thigh while sitting, to performing gestures side of the leg while standing. As seen in Figure 32, participants were confused by this, often swiping down when they should swipe up (yellow dots on the bottom), and vice versa (red dots on the top). The inconsistency of participants' default "up" swipe direction and the resulting errors from it suggest that a calibration setting may be beneficial to CountMarks. Future implementations may want to let the user choose their desired swipe orientation.

Figure 32: Location of the end point of incorrect Expert selections

Figure 32 also shows straight lines of errors from targets in diagonal directions. These lines of colour show that participants often selected just outside the boundary of the intended target. Moreover, they mostly made errors on only one side of the target. This is likely a result of participants swiping in an arch. Because our fingers, hands, and arms move on joints, our movements are naturally curved rather than perfectly straight. We show that these curved movements are relatively consistent across right-handed users. Participants appear to finish their swipe gesture closer to the horizontal axis of selection or in a clockwise direction if the target is directly above or below on the vertical axis. This is supported by fewer errors being shown for targets that were on the horizontal axis (left and right directions). We would recommend that the target areas for selection be modified to account for this type of movement by decreasing the size of the left and right selection areas and expanding the selection areas for the diagonal swipe gestures.

Additionally, the top and bottom regions may be expanded slightly in the clockwise direction.

5.6.2 User Stance and Device Positioning

In this study we evaluated marking menu selections in two previously unexamined domains: while standing vs. walking, and with the input device handheld vs. leg-mounted. The first main surprise was that walking only mildly affected performance. Walking slowed reaction times by only 0.060s while showing no significant effect on execution speeds. This shows that users are able to perform CountMarks almost as quickly while in motion as while standing still. Since CountMarks is designed to be used with a mobile HMD this is very important.

The real problem is that Walking dropped the accuracy of selection by 6% on average across all conditions. This is primarily caused by an interaction effect of user stance and device position. Accuracy differed by only 2% between user stance conditions when the input device was in the participant's hands but dropped 17% when it was mounted to the leg. Leg interaction while standing had an accuracy of 92.5% in the Novice task which remains in the range of other marking menu techniques (Table 2), but the accuracy drops considerably to 75.9% when walking. We anticipated lower accuracy for the Leg/Walking condition due to device location being on the body part used for motion. This motion causes the plane of the interaction surface to be constantly changing. As the leg moves while walking, the touch surface shifts its orientation with the leg such the horizontal axis of the phone is not the same at all times. Therefore, swipe directions may need to be relative to the orientation of the device in motion. Eight participants specifically mentioned the difficulty of coordinating hand and leg movements when making selections. Despite this difficulty and the possible finger recognition problems described above, we were surprised with how accurate user were able to perform Leg/Walking gestures. We propose that future work investigate adjusting the user's swipe direction based on the input device's change in orientation while walking. We hypothesize that by fixing issues of orientation during leg movement, calibrating for users preconceived notion of "up", and adjusting selection areas and finger count recognition that leg interaction may become a viable option for mobile marking menu interactions. While some of the study's limitations are mentioned above, we note that similar to Study 1 the results are limited by the unrealistic use case of the Novice selection task, and by our participants being primarily young students which are not representative of the general population. We will conclude this thesis with examples of how CountMarks may be implemented into existing user interfaces

Chapter 6: Example Applications and Conclusions

6.1 Example Applications

Our studies show that CountMarks is a viable variation of marking menus, improving upon metrics such as selection speed, search accuracy and in some cases selection accuracy. This is likely because CountMarks supports greater menu breadth rather than depth. With this greater breadth, designers may not need to trade off minimizing the user's memory load and error prevention in order to make shortcuts available to their users. We demonstrate this by implementing CountMarks in a mock Netflix app (Figure 33). This app is designed to be viewed on an HMD or a distant display like a TV. Using regular marking menus to select a movie in a Netflix app would be very difficult. The user would have to first identify the desired movie's genre through lists of 8 genres at a time and would then have to drill down into that genre's movie list to identify their selection. The user would also need a way to move back up the menu hierarchy if they changed their mind. Regardless of the implementation, it would most likely require a drastic redesign of Netflix's current interface.

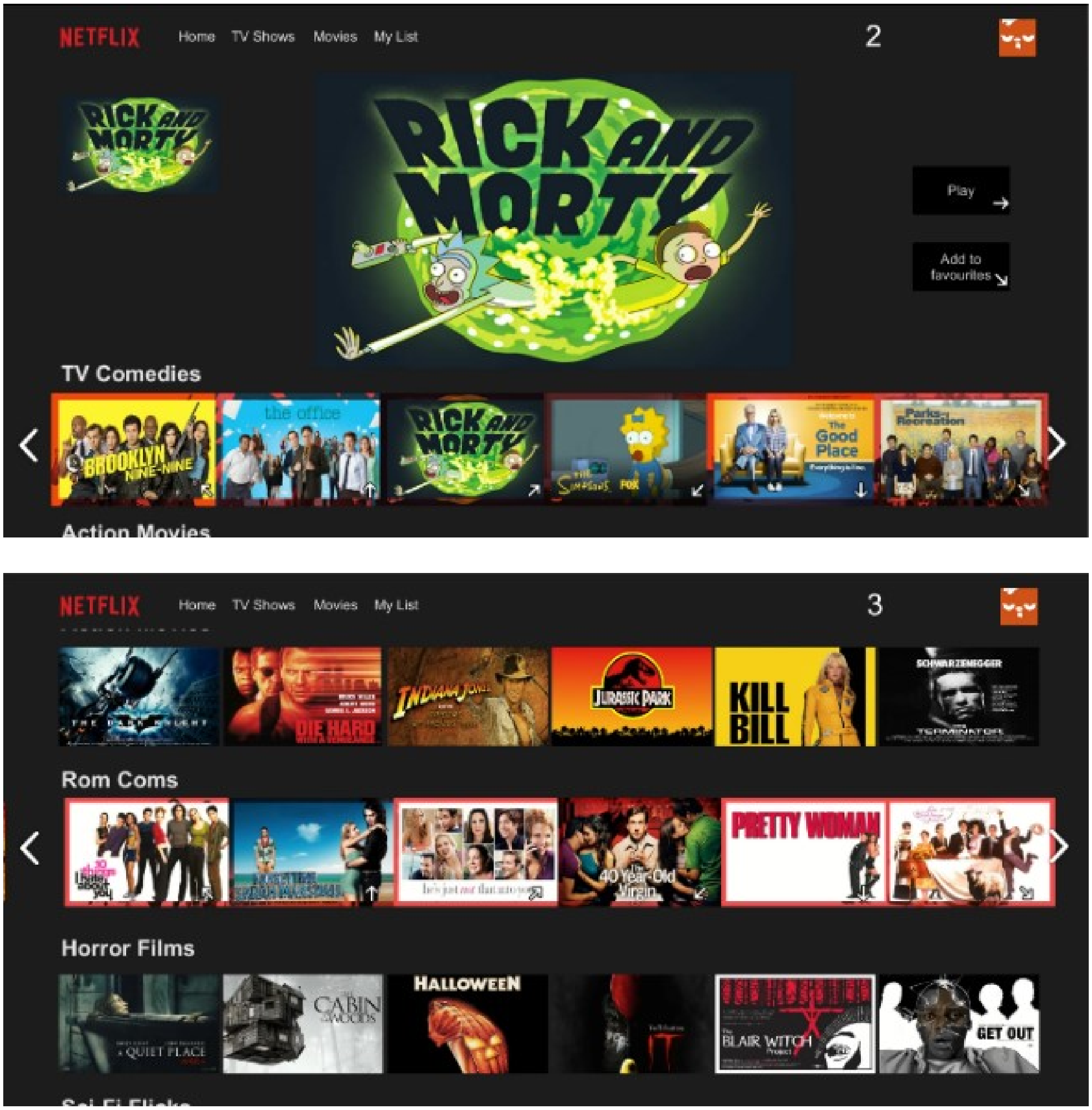

Figure 33: Two screen shots of a mock Netflix demo with CountMarks implemented for selection. From top to bottom, (a) shows two fingers held beginning a selection, (b) shows three fingers held down beginning a selection.

By using CountMarks, we can keep the layout of the app the same and merely change the interaction method. The user can scroll vertically by sliding one finger on the screen. Touching two or more fingers on the screen activates CountMarks to select from different movie genres. Touching two fingers activates the top row, three fingers activate the middle, and four fingers activate the bottom row. Upon activating a row in this fashion, it becomes highlighted, and small arrows appear in the corner of the movie images corresponding to the swipe direction required to select that movie. Swiping left or right with the desired number of fingers will horizontally scroll the specified row. Finally, a double tap toggles the selection mode to focus on the top bar of items where the user can select settings and other options.

This mockup uses the MenuCount variation of CountMarks, allowing users to place a desired number of fingers to activate a desired menu (in this case genre). This is easily implemented for Netflix because users must first decide what genre they would like to watch before selecting an item from that menu. This works particularly well because the menus (genres) are neatly arranged in the UI. The limitation of this is that MenuCount can only select from 4 menus or semantic groupings at a time before adding complexity with a mode change gesture. In this application, the mode change is used to access secondary selectable items such as settings, search and account info.

Applications that have a greater number of menus but fewer items to select in each menu may benefit more from the ItemCount variation of CountMarks. An example of this could be a design application with a toolbar that is broken into 8 sections, where each section has only a few tools that are commonly used. The number of fingers used for each selection would determine which item (tool) is selected, and the user swipes in the direction that corresponds to that tool's section.

The biggest challenge for future implementations of CountMarks is identifying how CountMarks can be generalized to map onto any existing interface. The internet, and applications in general, have been completely redesigned over the last decade and a half to accommodate the smaller screen size of smartphones. While it is unclear how application interfaces will change with the future for HMDs, we aim to design CountMarks to be flexible enough to be implemented onto these new (as well as current) interfaces. The Netflix app above shows that CountMarks is easily implemented for interfaces with structured sets of items. But not all interfaces contain easily grouped or structured content. Wikipedia, for example, presents dozens of clickable links or photos at once (Figure 34). These links are placed wherever the editor deems necessary and appear unorganized. Future work includes exploring how to generalize the mapping of CountMarks to webpages and apps that are dense with selectable items like this. This should be done in a consistent way such that users intuitively know how to use CountMarks to interact with any new user interface.

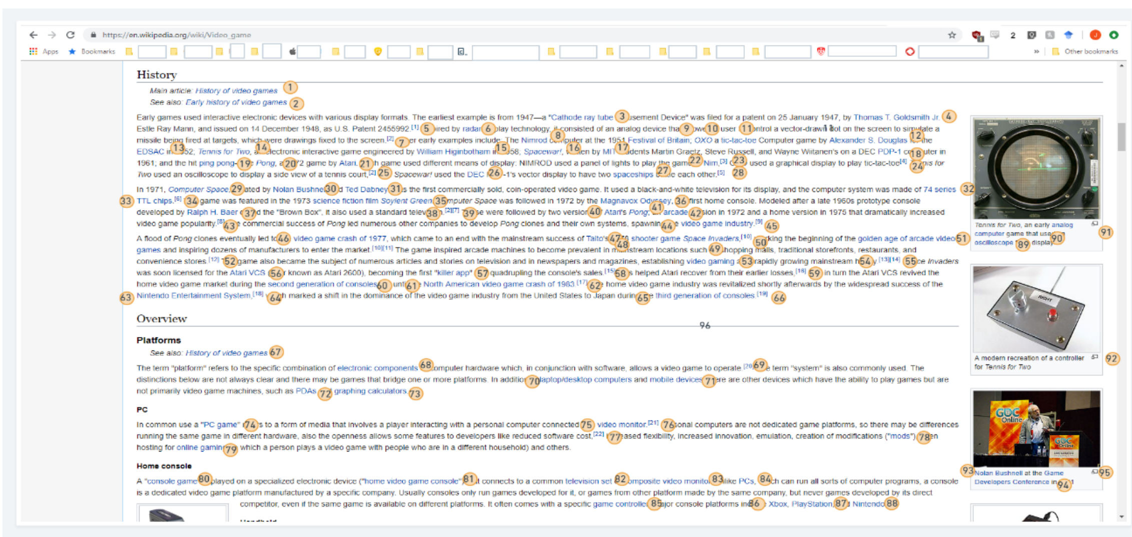

Figure 34: An image of a Wikipedia page showing how many clickable links can fit on a screen at one point

6.2 Conclusion

In this thesis, we designed, implemented and evaluated a novel multi-touch marking menu technique designed for mobile interactions with HMDs. This work was inspired by previous research on Count Menus, marking menus and on-body interaction, to create an interaction technique that is designed to be fast, accurate, ergonomic and appropriate for use in a public setting.

We conducted two experiments to evaluate CountMarks. The first compared two variations of CountMarks to existing marking menus. The second experiment evaluated the efficacy of CountMarks in various mobile use cases. CountMarks overcomes the limited breadth of traditional marking menus by allowing 32 items in a single depth and up to 64 items with a mode shifting double tap. Our findings show that the MenuCount variation of CountMarks offers quicker selection and improved search accuracy than multi-stroke marking menus at the cost of a minor decrease in selection accuracy.

Using the MenuCount variation, we then tested CountMarks in multiple use cases. We had participants use CountMarks to perform Novice and Expert selections in a Microsoft HoloLens while standing and walking and with the phone held in the hand and attached to the leg. Expert selection showed a much lower accuracy than Novice selection, but we believe this is due to hardware and software limitations of finger recognition and that by solving this problem the error rate should be reduced to an acceptable level. Walking showed only a minor effect on selection time but did cause a significant decrease in accuracy when the input device was at the leg. Leg interactions in general were slower and more prone to errors than handheld interactions. Overall, we show that Novice selection using CountMarks outperforms even Expert selection from most other marking menu variations. We further propose solutions for reducing error rates by fixing finger recognition issues, adjusting the sizes of the selection areas and calibrating selection orientation for participants. With these adjustments, we anticipate Expert selections will approach the accuracy of Novice selections while being significantly faster and that leg interaction may still be viable for mobile marking menu interaction. Future work should also explore the ergonomic and social acceptability of CountMarks.

We also demonstrate how by increasing the breadth of Marking menu interactions, designers can more easily implement CountMarks into user interfaces. We provide an example of how CountMarks could be applied to a Netflix application without needing to redesign the app.

Altogether we contribute an improved marking menu style of selection using multiple fingers. We provide empirical evidence into the strengths and weakness of performing these interactions while in mobile settings and with different device input locations. Expanding upon well-established marking menus, we believe CountMarks is a viable alternative for mobile interaction with HMDs that would otherwise use in-air gestures, voice or joystick-based controls.

Appendices

Appendix A

A.1 Consent Form for CountMarks vs Marking menu Study

Carleton UNIVERSITY
Canada's Capital University

Consent Form

Title: CountMarks: Indirect touch input for mobile computing applications

Date of ethics clearance: To be determined by the REB (as indicated on the clearance form)
Ethics Clearance for the Collection of Data Expires: To be determined by the REB (as indicated on the clearance form)

I ____, choose to participate in a study on touch interaction for head-mounted displays with movement. I acknowledge that this study aims to assess the strength and the weakness of using the CountMarks technique at the hip or hand level. **The researcher for this study is Masters' student Jordan Pollock**. He is working under the supervision of Dr. Robert Teather **in The School of Information Technology.**

This study will take one hour in total. It involves a pre-test demographic questionnaire, an activity (30-45 minute AR exposure including up to 25 minutes of walking) and a short post-test interview.

If you feel uncomfortable, tired or nauseous you have the right to end your participation in the study at any time during the session, for any reason. Simply tell the researcher that you want to end the session. If you withdraw from the study, all information you have provided will be immediately destroyed. We will just mention that one of the participants could not finish the test. You may not withdraw from the study after leaving the examination room.

As a token of appreciation, you will receive a $10. This is yours to keep, even if you withdraw from the study.

All responses and data will be kept anonymous. All research data and notes will be kept on a password protected computer of the researchers. Research data will only be accessible by the researchers and the research supervisor. This data may be used in publications and future research as well as for presentations and teaching purposes. If used, the data will remain anonymous.

Once the project is completed, all research data will be securely destroyed.

Page 1 of 2

**This document has been printed on both sides of a single sheet of paper.
Please retain a copy of this document for your records.**

Once the project is completed, all research data will be securely destroyed. Electronic data will be erased one year after collection and hard copies will be shredded after they are transcribed and analyzed.

The ethics protocol for this project was reviewed by the Carleton University Research Ethics Board with CUREB-B Clearance #110191 to carry out the research. Should you have any ethical concerns with the study, please contact Dr. Bernadette Campbell, Chair, Carleton University Research Ethics Board-B (by phone: 613-520-2600 ext. 4085 or by email: ethics@carleton.ca). For all other questions about the study, please contact the researcher.

Researchers contact information:

Jordan Pollock
School of Information Technology
Carleton University
JordanPollock@cmail.carleton.ca

Supervisor contact information:
Dr. Robert Teather
School of Information Technology
Carleton University
613-520-2600x4176
rob.teather@carleton.ca

________________________	______________
Signature of participant	Date
________________________	______________
Signature of a researcher	Date

Page 2 of 2

This document has been printed on both sides of a single sheet of paper.
Please retain a copy of this document for your records.

A.2 Demographic Form for CountMarks vs Marking menu Study

Participant number *

Your answer

Age

Your answer

Gender

◯ Female

◯ Male

◯ Prefer not to say

◯ Other: ____________

Are you right handed or left handed?

◯ Right

◯ Left

◯ Both

Do you have any visual impairments?

◯ No

◯ Glasses/Contacts

◯ Other: ____________

Do you have any motor impairments of the hand or arm?

◯ No

◯ Yes

SUBMIT

Never submit passwords through Google Forms.

A.3 Questionnaire for CountMarks vs Marking menu Study

Which selection technique was the quickest to perform?

- MenuCount
- ItemCount
- Marking Menu

Which selection technique was the easiest to learn?

- MenuCount
- ItemCount
- Marking Menu

Which selection technique was the most tiring to perform?

- MenuCount
- ItemCount
- Marking Menu

Which selection technique was your favourite?

- MenuCount
- ItemCount
- Marking Menu

NEXT

A.4 Consent Form for CountMarks Position and Movement Study

Carleton UNIVERSITY
Canada's Capital University

Consent Form

Title: CountMarks: Multi-touch marking menus for head-mounted and distant displays

Date of ethics clearance: To be determined by the REB (as indicated on the clearance form)
Ethics Clearance for the Collection of Data Expires: To be determined by the REB (as indicated on the clearance form)

I ____, choose to participate in a study on Marking Menu selection. I acknowledge that this study aims to assess the strength and the weakness of using the CountMarks technique compared to an existing menu selection technique. **The researcher for this study is Masters' student Jordan Pollock**. He is working under the supervision of Dr. Robert Teather **in The School of Information Technology.**

This study will take one hour in total. It involves a pre-test demographic questionnaire, an activity (30-45 minute of selection tasks) and a short post-test interview.

If you feel uncomfortable or tired, you have the right to end your participation in the study at any time during the session, for any reason. Simply tell the researcher that you want to end the session. If you withdraw from the study, all information you have provided will be immediately destroyed. We will just mention that one of the participants could not finish the test. You may not withdraw from the study after leaving the examination room.

As a token of appreciation, you will receive a $10. This is yours to keep, even if you withdraw from the study.

All responses and data will be kept anonymous. All research data and notes will be kept on a password protected computer of the researchers. Research data will only be accessible by the researchers and the research supervisor. This data may be used in publications and future research as well as for presentations and teaching purposes. If used, the data will remain anonymous.

Page 1 of 2

This document has been printed on both sides of a single sheet of paper.
Please retain a copy of this document for your records.

Electronic data will be erased one year after collection and hard copies will be shredded after they are transcribed and analyzed.

The ethics protocol for this project was reviewed by the Carleton University Research Ethics Board with CUREB-B Clearance #110191 to carry out the research. Should you have any ethical concerns with the study, please contact Dr. Bernadette Campbell, Chair, Carleton University Research Ethics Board-B (by phone: 613-520-2600 ext. 4085 or by email: ethics@carleton.ca). For all other questions about the study, please contact the researcher.

Researchers contact information:

Jordan Pollock
School of Information Technology
Carleton University
JordanPollock@cmail.carleton.ca

Supervisor contact information:
Dr. Robert Teather
School of Information Technology
Carleton University
613-520-2600x4176
rob.teather@carleton.ca

_________________________ _______________
Signature of participant Date

________________________ _______________
Signature of a researcher Date

Page 2 of 2

This document has been printed on both sides of a single sheet of paper.
Please retain a copy of this document for your records.

A.5 Demographic Form for CountMarks Position and Movement Study

Participant number

Your answer

Age *

Your answer

Height

Your answer

Gender

- Female
- Male
- Prefer not to say
- Other:

Treadmill speed

- 1.5
- 2.0
- 2.5

Do you have any visual or motor impairments that aren't corrected for? *

- Yes
- No
- Maybe
- Other:

Do you have any visual or motor impairments that aren't corrected for? *

- Yes
- No
- Maybe
- Other:

When seeing the up arrow did you before starting the experiment did you...

- Swipe towards your head
- Swipe towards your feet
- Swipe towards the front of the phone
- Swipe towards the back of the phone

Have you participated in a CountMarks study before?

- Yes
- No

Do you text or look at apps on your phone while walking? *

- Always
- Often
- Sometimes
- Rarely
- Never

A.6 Questionnaire for CountMarks Position and Movement Study

Your experience using CountMarks while standing still

This is your opportunity to let us know what you thought about your experience using CountMarks

How comfortable was interacting with the display while *Standing* and *Holding the phone*?

	1	2	3	4	5	
Very uncomfortable	○	○	○	○	○	Very comfortable

How easy was interacting with the display while *Standing* and *holding the phone*?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

How comfortable was interacting with the display while *standing* with the *phone on your leg*?

	1	2	3	4	5	
Very uncomfortable	○	○	○	○	○	Very comfortable

How easy was interacting with the display while *standing* with the *phone on your leg*?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

Any comments about your interactions using CountMarks while standing?

Your answer

Your experience using CountMarks while walking

This is your opportunity to let us know what you thought about your experience using CountMarks

How comfortable was interacting with the display while *walking* and *holding the phone*?

	1	2	3	4	5	
Very uncomfortable	○	○	○	○	○	Very comfortable

How easy was interacting with the display while *walking* and *holding the phone*?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

How comfortable was interacting with the display while *walking* with the *phone on your leg*?

	1	2	3	4	5	
Very uncomfortable	○	○	○	○	○	Very comfortable

How easy was interacting with the display while *walking* with the *phone on your leg*?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

Any comments about your interactions using CountMarks while walking?

Your answer

References

[1] R. R. Schaller, “Moore’s law: past, present and future,” *IEEE Spectr.*, vol. 34, no. 6, pp. 52–59, 1997.

[2] I. E. Sutherland, “A head-mounted three dimensional display,” in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, 1968, pp. 757–764.

[3] Y.-C. Tung *et al.*, “User-Defined Game Input for Smart Glasses in Public Space,” *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst. - CHI ’15*, pp. 3327–3336, 2015.

[4] J. D. Hincapié-Ramos, X. Guo, P. Moghadasian, and P. Irani, “Consumed endurance,” *Proc. 32nd Annu. ACM Conf. Hum. factors Comput. Syst. - CHI ’14*, no. February, pp. 1063–1072, 2014.

[5] A. Cockburn, P. Quinn, C. Gutwin, G. Ramos, and J. Looser, “Air pointing: Design and evaluation of spatial target acquisition with and without visual feedback,” *Int. J. Hum. Comput. Stud.*, vol. 69, no. 6, pp. 401–414, 2011.

[6] M. Weiser, “The Computer for 21 st Century,” *Sci. Am. Ubicomp Pap.*, vol. 265, no. 3, p. 8, 1991.

[7] G. P. Kurtenbach, *The design and evaluation of marking menus.* University of Toronto Toronto, 1993.

[8] J. Callahan, D. Hopkins, M. Weiser, and B. Shneiderman, “An empirical comparison of pie vs. linear menus,” *Proc. SIGCHI Conf. Hum. factors Comput. Syst. - CHI ’88*, pp. 95–100, 1988.

[9] G. Kurtenbach and W. Buxton, “The limits of expert performance using hierarchic

marking menus," *Proc. SIGCHI Conf. Hum. factors Comput. Syst. - CHI '93*, pp. 482–487, 1993.

[10] K. Snowberry, S. R. Parkinson, and N. Sisson, "Computer display menus," *Ergonomics*, vol. 26, no. 7, pp. 699–712, 1983.

[11] R. Molich and J. Nielsen, "Improving a human-computer dialogue," *Commun. ACM*, vol. 33, no. 3, pp. 338–349, 1990.

[12] J. Gugenheimer, D. Dobbelstein, C. Winkler, G. Haas, and E. Rukzio, "FaceTouch: Enabling Touch Interaction in Display Fixed UIs for Mobile Virtual Reality," *Proc. ACM Symp. User Interface Softw. Technol. (UIST '16)*, pp. 49–60, 2016.

[13] M. Serrano, B. M. Ens, and P. P. Irani, "Exploring the use of hand-to-face input for interacting with head-worn displays," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2014, pp. 3181–3190.

[14] G. Kurtenbach and W. Buxton, "User learning and performance with marking menus," *Proc. SIGCHI Conf. Hum. factors Comput. Syst. Celebr. Interdepend. - CHI '94*, pp. 258–264, 1994.

[15] G. Bailly, E. Lecolinet, and L. Nigay, "Visual Menu Techniques," *ACM Comput. Surv.*, vol. 49, no. 4, pp. 1–41, 2016.

[16] S. Zhao and R. Balakrishnan, "Simple vs. compound mark hierarchical marking menus," *Proc. 17th Annu. ACM Symp. User interface Softw. Technol. - UIST '04*, p. 33, 2004.

[17] J. I. Kiger, "The depth/breadth trade-off in the design of menu-driven user

interfaces," *Int. J. Man. Mach. Stud.*, vol. 20, no. 2, pp. 201–213, 1984.

[18] G. Bailly, E. Lecolinet, and L. Nigay, "Flower menus," *Proc. Work. Conf. Adv. Vis. interfaces - AVI '08*, p. 15, 2008.

[19] S. Zhao, M. Agrawala, and K. Hinckley, "Zone and polygon menus: using relative position to increase the breadth of multi-stroke marking menus," *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, pp. 1077–1086, 2006.

[20] G. P. Kurtenbach, "Display and control of menus with radial and linear portions." Google Patents, 20-Jul-1999.

[21] G. J. Lepinski, T. Grossman, and G. Fitzmaurice, "The design and evaluation of multitouch marking menus," *Proc. 28th Int. Conf. Hum. factors Comput. Syst. - CHI '10*, p. 2233, 2010.

[22] G. Bailly, R. Walter, J. Müller, T. Ning, and E. Lecolinet, "Comparing free hand menu techniques for distant displays using linear, marking and finger-count menus," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6947 LNCS, no. PART 2, pp. 248–262, 2011.

[23] A. Kulshreshth and J. J. LaViola, "Exploring the usefulness of finger-based 3D gesture menu selection," *Proc. 32nd Annu. ACM Conf. Hum. factors Comput. Syst. - CHI '14*, pp. 1093–1102, 2014.

[24] G. Bailly, J. Müller, and E. Lecolinet, "Design and evaluation of finger-count interaction: Combining multitouch gestures and menus," *Int. J. Hum. Comput. Stud.*, vol. 70, no. 10, pp. 673–689, 2012.

[25] G. Ren and E. O'Neill, "3D Marking menu selection with freehand gestures,"

IEEE Symp. 3D User Interfaces 2012, 3DUI 2012 - Proc., pp. 61–68, 2012.

[26] S. Zhao and R. Balakrishnan, “Simple vs. compound mark hierarchical marking menus,” *Proc. 17th Annu. ACM Symp. User interface Softw. Technol. - UIST ’04*, vol. 6, no. 2, p. 33, 2004.

[27] B. Shneiderman, “Direct manipulation: A step beyond programming languages (excerpt),” *Readings Human-Computer Interact. A Multidiscip. Approach*, pp. 461–467, 1987.

[28] D. M. Lane, H. A. Napier, S. C. Peres, and A. Sandor, “Hidden costs of graphical user interfaces: Failure to make the transition from menus and icon toolbars to keyboard shortcuts,” *Int. J. Hum. Comput. Interact.*, vol. 18, no. 2, pp. 133–144, 2005.

[29] J. M. Carroll and M. B. Rosson, “Paradox of the active user,” 1987.

[30] A. Cockburn, C. Gutwin, J. Scarr, and S. Malacria, “Supporting Novice to Expert Transitions in User Interfaces,” *ACM Comput. Surv.*, vol. 47, no. 2, pp. 1–36, 2014.

[31] L. Findlater, K. Moffatt, J. McGrenere, and J. Dawson, “Ephemeral adaptation: The use of gradual onset to improve menu selection performance,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2009, pp. 1655–1664.

[32] O. Bau and W. E. Mackay, “OctoPocus: a dynamic guide for learning gesture-based command sets,” in *Proceedings of the 21st annual ACM symposium on User interface software and technology*, 2008, pp. 37–46.

[33] Y.-T. Hsieh, A. Jylhä, V. Orso, L. Gamberini, and G. Jacucci, “Designing a

Willing-to-Use-in-Public Hand Gestural Interaction Technique for Smart Glasses," *Proc. 2016 CHI Conf. Hum. Factors Comput. Syst. - CHI '16*, pp. 4203–4215, 2016.

[34] J. Rico and S. Brewster, "Usable gestures for mobile interfacess: Evaluating Social Acceptability," *Proc. 28th Int. Conf. Hum. factors Comput. Syst. - CHI '10*, p. 887, 2010.

[35] A. Lucero and A. Vetek, "NotifEye," *Proc. 11th Conf. Adv. Comput. Entertain. Technol. - ACE '14*, pp. 1–10, 2014.

[36] M. Koelle, M. Kranz, and M. Andreas, "Don ' t look at me that way ! – Understanding User Attitudes Towards Data Glasses Usage," *MobileHCI '15 Proc. 17th Int. Conf. Human-Computer Interact. with Mob. Devices Serv.*, pp. 362–372, 2015.

[37] C. Harrison, H. Benko, and A. D. Wilson, "OmniTouch," *Proc. 24th Annu. ACM Symp. User interface Softw. Technol. - UIST '11*, p. 441, 2011.

[38] M. Liu, M. Nancel, and D. Vogel, "Gunslinger : Subtle Arms-down Mid-air Interaction," pp. 63–71.

[39] M. Fukumoto and Y. Tonomura, "'Body coupled FingerRing': Wireless Wearable Keyboard," *Proc. SIGCHI Conf. Hum. factors Comput. Syst. - CHI '97*, pp. 147–154, 1997.

[40] X.-D. Yang, T. Grossman, D. Wigdor, and G. Fitzmaurice, "Magic Finger: Always-Available Input through Finger Instrumentation," *Proc. 25th Annu. ACM Symp. User interface Softw. Technol. - UIST '12*, pp. 147–156, 2012.

[41] C. G. L. Cao, M. Zhou, D. B. Jones, and S. D. Schwaitzberg, "Can surgeons think

and operate with haptics at the same time?," *J. Gastrointest. Surg.*, vol. 11, no. 11, pp. 1564–1569, 2007.

[42] T. Ni and P. Baudisch, "Disappearing mobile devices," *Proc. UIST 2009*, pp. 101–110, 2009.

[43] C. Harrison, D. Tan, and D. Morris, "Skinput," *Proc. 28th Int. Conf. Hum. factors Comput. Syst. - CHI '10*, p. 453, 2010.

[44] M. Weigel, A. S. Nittala, A. Olwal, and J. Steimle, "SkinMarks: Enabling Interactions on Body Landmarks Using Conformal Skin Electronics," *Proc. 2017 CHI Conf. Hum. Factors Comput. Syst. - CHI '17*, pp. 3095–3105, 2017.

[45] Y. Zhang, J. Zhou, G. Laput, and C. Harrison, "SkinTrack: Using the Body as an Electrical Waveguide for Continuous Finger Tracking on the Skin," *Proc. 2016 CHI Conf. Hum. Factors Comput. Syst. - CHI '16*, pp. 1491–1503, 2016.

[46] M. Weigel, V. Mehta, and J. Steimle, "More Than Touch : Understanding How People Use Skin as an Input Surface for Mobile Computing," *Proc. 32nd Annu. ACM Conf. Hum. factors Comput. Syst. - CHI '14*, pp. 179–188, 2014.

[47] C.-Y. Wang *et al.*, "PalmGesture: using palms as gesture interfaces for eyes-free input," *Proc. 17th Int. Conf. Human-Computer Interact. with Mob. Devices Serv.*, pp. 217–226, 2015.

[48] L. Chan *et al.*, "FingerPad: Private and Subtle Interaction Using Fingertips," *Proc. 26th Annu. ACM Symp. User interface Softw. Technol. - UIST '13*, pp. 255–260, 2013.

[49] K.-Y. Chen, K. Lyons, S. White, and S. Patel, "uTrack: 3D Input Using Two Magnetic Sensors," *Proc. 26th Annu. ACM Symp. User Interface Softw. Technol.*,

pp. 237–244, 2013.

[50] H.-L. (Cindy) Kao, A. Dementyev, J. A. Paradiso, and C. Schmandt, "NailO: Fingernails as an Input Surface," *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst. - CHI '15*, pp. 3015–3018, 2015.

[51] J. Lien *et al.*, "Soli: Ubiquitous Gesture Sensing with Millimeter Wave Radar," *ACM Trans. Graph. Artic.*, vol. 35, no. 10, pp. 1–19, 2016.

[52] D. Ashbrook, P. Baudisch, and S. White, "Nenya," *Proc. 2011 Annu. Conf. Hum. factors Comput. Syst. - CHI '11*, p. 2043, 2011.

[53] L. Chan, Y.-L. Chen, C.-H. Hsieh, R.-H. Liang, and B.-Y. Chen, "CyclopsRing: Enabling Whole-Hand and Context-Aware Interactions Through a Fisheye Ring," *Proc. 28th Annu. ACM Symp. User Interface Softw. Technol. - UIST '15*, no. 1, pp. 549–556, 2015.

[54] R. Shilkrot, J. Huber, J. Steimle, S. Nanayakkara, and P. Maes, "Digital Digits: A Comprehensive Survey of Finger Augmentation Devices," *ACM Comput. Surv.*, vol. 48, no. 2, p. 30:1--30:29, 2015.

[55] M. Weigel, T. Lu, G. Bailly, A. Oulasvirta, C. Majidi, and J. Steimle, "iSkin: Flexible, Stretchable and Visually Customizable On-Body Touch Sensors for Mobile Computing," *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst. - CHI '15*, pp. 2991–3000, 2015.

[56] S. Zhao, P. Dragicevic, M. Chignell, R. Balakrishnan, and P. Baudisch, "earPod: Eyes-free Menu Selection using Touch Input and Reactive Audio Feedback," *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, p. 1395, 2007.

[57] S. Sridhar, A. Markussen, A. Oulasvirta, C. Theobalt, and S. Boring,

"WatchSense," *Proc. 2017 CHI Conf. Hum. Factors Comput. Syst. - CHI '17*, pp. 3891–3902, 2017.

[58] D. Dobbelstein, P. Hock, and E. Rukzio, "Belt: An Unobtrusive Touch Input Device for Head-worn Displays," *Proc. ACM CHI'15 Conf. Hum. Factors Comput. Syst.*, vol. 1, pp. 2135–2138, 2015.

[59] T. S. Saponas, C. Harrison, and H. Benko, "PocketTouch," *Proc. 24th Annu. ACM Symp. User interface Softw. Technol. - UIST '11*, no. Figure 1, p. 303, 2011.

[60] M. Pielot, T. Hesselmann, W. Heuten, A. Kazakova, and S. Boll, "PocketMenu: Non-Visual Menus for Touch Screen Devices," *Proc. Int. Conf. Human-computer Interact. with Mob. devices Serv. (MobileHCI '13)*, pp. 327–330, 2012.

[61] M. Weigel and J. Steimle, "DeformWear: Deformation Input on Tiny Wearable Devices," *Proc.ACMInteract.Mob. WearableUbiquitousTechnol*, vol. 12, no. 2, pp. 1–23, 2017.

[62] S. H. Yoon, K. Huo, and K. Ramani, "Wearable textile input device with multimodal sensing for eyes-free mobile interaction during daily activities," *Pervasive Mob. Comput.*, vol. 33, pp. 17–31, 2016.

[63] J. Zheng, X. Bi, K. Li, Y. Li, and S. Zhai, "M3 Gesture Menu," *Proc. 2018 CHI Conf. Hum. Factors Comput. Syst. - CHI '18*, pp. 1–14, 2018.

[64] T. S. Saponas, D. S. Tan, D. Morris, R. Balakrishnan, J. Turner, and J. A. Landay, "Enabling always-available input with muscle-computer interfaces," *Proc. 22nd Annu. ACM Symp. User interface Softw. Technol. - UIST '09*, p. 167, 2009.

[65] J. Gong, X.-D. Yang, and P. Irani, "WristWhirl: One-handed Continuous Smartwatch Input using Wrist Gestures," *Proc. 29th Annu. Symp. User Interface*

Softw. Technol., pp. 861–872, 2016.

[66] G. Bailly, E. Lecolinet, and L. Nigay, “Wave Menus: Improving the Novice Mode of Hierarchical Marking Menus,” *Interact*, pp. 475–488, 2007.