Designing a Foot Input System for Productive Work at a Standing Desk

by

William Saunders

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2015

AUTHOR'S DECLARATION

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

In this thesis, I describe the work that I carried out with my supervisor, Professor Daniel Vogel, over the course of my Master's degree. Content of this thesis is included from conference submissions that I co-wrote with Daniel Vogel. The study described in Chapter 3 is the subject of a submission to Graphics Interfaces 2015, and is pending review at the time of this writing. The system described in Chapter 4 is the subject of a paper to be submitted to UIST 2015. Content from these submissions was used in these chapters. All chapters of the thesis may include some content from these submissions, though I made extensive revisions and wrote new content to make this thesis a cohesive whole.

I made the major contribution to the research described in this thesis. I proposed the original program of research, performed most of the literature search, conducted the experiments, wrote the experimental code, performed all statistical analysis, shot original video for the video figures, wrote the first draft of the majority of content for both submissions, and participated in further revisions. Daniel Vogel advised on all aspects of the study, wrote some content, created original figures, edited the video, and revised the conference submissions.

Abstract

In this thesis we present Tap-Kick-Click, a foot interaction system for controlling common desktop applications. This system enables computer workers to take healthy and productive breaks from using a keyboard and mouse and demonstrates foot interaction techniques which could be applied in other contexts. Our work supplements the existing literature on foot based interaction, as no published work has combined foot input with a standing desk or attempted control of conventional desktop applications.

We describe two experiments to investigate questions about the human performance characteristics of foot input relevant to our application which were unanswered in the existing literature. These experiments investigated the effect of target size, direction and distance; the difference between dominant and non-dominant foot; the use of tapping and kicking interaction; and the impact of displaying or hiding a foot cursor. Based on our results we present a set of design guidelines including a suggested minimum target size; a recommendation to ignore foot dominance; and a preference ranking for direction and foot action.

These design guidelines informed the design of Tap-Kick-Click, which we describe in detail. It uses a sensing technique using a Microsoft Kinect depth camera and a pair of augmented slippers capable of robustly sensing foot position, kicking and tapping. The primary interaction technique is based on combinations of foot action and directional tapping in a low-density target layout, supported by feedback and instructions presented in an always visible sidebar. This technique is supplemented with a system for selecting elements in a GUI, a high-density target layout for selecting items from a menu, and a help screen. We illustrate the usefulness of Tap-Kick-Click by describing how it can be used to control a web browser, a citation manager and a debugger.

Finally, we present the results of a study conducted to evaluate whether new users could learn and use the system in a web browser context. The study demonstrated that users are successfully able to learn and use the system, along with providing areas for improvement.

Acknowledgements

I would like to express my gratitude to everyone who helped me along my way to completing this thesis and my degree.

First, I would like to thank my supervisor Daniel Vogel, who was an integral part of all of the research I discuss here. You believed in my research goal and were willing to support me in seeing it to its conclusion. You guided me through the process of designing and conducting research, showed me the ins and outs of the HCI world, and helped bring clarity to the expression of my ideas.

I also want to thank the other faculty and students of the HCI research group. Mike Terry, Edward Lank, Filip Krynicki, Valerie Sugarman, Adam Fourney, Jeff Avery, Yuexing Luo, Mingyu Liu and everyone else in the lab, I learned many things from you and always found you interested in my research and eager to help. I'm grateful to have had the chance to share the experiences of academia with you.

I would also like to everyone who came into the lab, put on strange footwear with markers and blinking lights, listened to my instructions, and danced around in front of my computer to participate in one of my studies. I literally couldn’t have done my research without you!

Finally, I would like to thank my family, for their support through both easy and challenging times in my degree: my parents Eric and Kathy for your constant love and encouragement; and my brother Daniel, for all those times that you pulled me away from my work to share a laugh or a moment of imagination, and gain much needed perspective.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Motivation

In recent years, the term “sitting disease” has appeared in the popular media [46,47] to describe the negative health impacts of a modern lifestyle that involves spending too much time sitting down at home and at work. A meta-analysis of six studies involving half a million adults published in PLOS ONE [6] found that sitting time increased all-cause mortality, even when physical activity was taken into account, and that 10 hours a day of sitting increased mortality risk of adults by 34%. This represents a problem for the knowledge-worker economy based around using computers for the majority of the working day. One solution to this problem is the standing desk, designed to hold the keyboard, monitor and mouse at a height appropriate for interaction while the user is standing. Research has suggested that using a standing desk can increase knowledge-worker health and productivity [13].

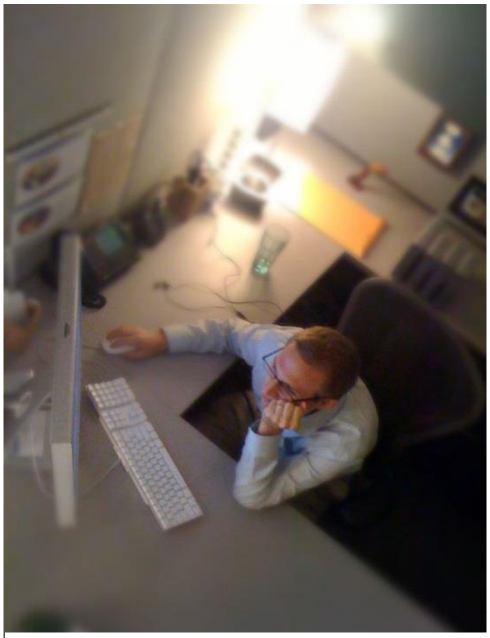


Figure 1-1: A desk-bound knowledge worker.
Photo By Benjamin Thompson (Flickr: Benjamin at Work) [CC BY-SA 2.0 (https://creativecommons.org/licenses/by-sa/2.0/)], via Wikimedia Commons

However, there is significant room for improvement in the standing desk concept. Computer work, especially when undertaken with poor posture, is a significant risk factor for some types of Repetitive Strain Injuries (RSI) or posture related muscle pain [22]. This is a widespread problem that is worth addressing: one study of Danish workers [2] found that 40% of white-collar workers had neck, shoulder, wrist or hand pain, and that this pain was a risk factor for long term sickness absence. Since a keyboard and mouse are used with a traditional standing desk, these health issues are still present. Taking small breaks from computer work can help [26], but these breaks are typically operationalized as simple reminders to stretch [16] or short periods of game play [28]. However, this task switching from work to break and back to work can be distracting and disruptive to productivity. Additionally, standing has been found to only provide modest increases in heart rate and energy expenditure [40]. Adding movement to use of a standing desk would provide even more physical activity.

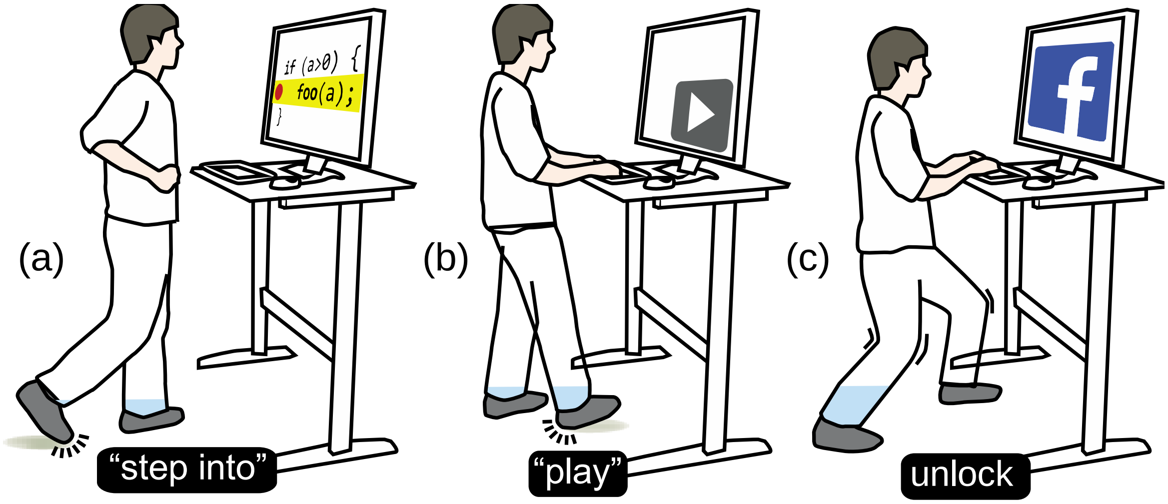


Figure 1-2. Tap-Kick-Click
Tap-Kick-Click uses: (a) physically active and productive "foot input only" break; (b) increased physical activity using feet to augment mouse and keyboard; (c) mildly uncomfortable foot positions to discourage cyberslacking.

In this thesis we describe *Tap-Kick-Click*, a set of interaction techniques to combine foot input with a standing desk to seamlessly enable productive "foot input only" healthy and active breaks (Figure 1-2a) and create opportunities for increased physical activity by using feet to augment mouse and keyboard input (Figure 1-2b). As a further probe into productive physical exertion, we also demonstrate how mildly uncomfortable foot positions could be used to discourage cyberslacking (time spent on distracting and unproductive websites) (Figure 1-2c).

The primary two-foot input vocabulary for *Tap-Kick-Click* uses discrete taps (with a toe, heel, or whole foot) and kicks, aimed at virtual targets arranged in a semicircular array around each foot. A low target density is used for eyes-free control and a high target density is used when the user can fully rely on indirect "foot cursor" control to select among many simultaneous actions. Our work significantly builds upon a work-in progress by Meyers et al. [27] that used a Dance Dance Revolution (DDR) game mat to sort email and photos using custom applications at a large display. We focus on a more general standing desk usage context, and we enable foot input with standard, unaltered desktop applications using a more expressive foot input vocabulary. Our design provides a balance between making interaction physically active; easy to learn and use; and reasonably efficient. This system also provides a new window into foot input research, as it enables a foot input system to be used for a wider variety of real-world tasks than have previously been studied.

To illustrate our system, this is a sample usage scenario for academic research:

Jane enters a paper search term with the keyboard, and lifts her hands off the desk to stretch while she scrolls the search results with forward and backward toe taps. She continues working "away from the keyboard" by kicking forward to enter "click mode" where visible hyperlinks are decorated with icons to convey the short sequences of forward, side, and back taps to select them. Jane selects a paper link by performing a sequence, reads the abstract, and with a backward kick, adds the paper to the Zotero reference manager. She switches to Zotero using a right kick and forward left toe tap, then opens the downloaded PDF with a forward kick. Jane skims the PDF while scrolling with her feet like the webpage. Having taken a short physical break, she reaches again for the keyboard to enter notes. While typing, her music player starts playing an annoying song, so she skips it "in the background" with a forward whole foot tap. Having accomplished some research, Jane decides to check Facebook. To help reduce procrastination, her system forces her to stand in a lunge-like position while viewing certain sites. It is just enough to deter her from spending too much time on it and Jane returns shortly to work.

1.2 Empirical research

While we initially set out with goal in mind of creating a system for interacting with desktop applications at a standing desk, we needed to make many design choices appropriate for the intended application. However, when we examined the literature (which we describe in Chapter 2), we did not find a technique which exactly suited our intended application, so we set out to define a new interaction technique. We chose to focus on *discrete* interaction, where distinct commands such as tapping and kicking are mapped to single actions in the target applications, as opposed to *continuous* interaction, where interaction parameters such as velocity map to a continuous parameter of the interaction. Discrete interaction is similar to application control via keyboard shortcuts, and can be used to emulate continuous interaction by auto-repeating commands. We chose to use a *foot pointing* technique, where foot actions such as taps occur over a *target* (a region in space, like a button in a GUI interface) and the combination of foot action and region would determine the discrete action performed by the application. This provides a large set of commands that can be sent using a simple foot movement. Finally, we chose to use an *indirect input mapping* where the targets and a *foot cursor* representing the location of the user's foot would be displayed on a monitor. This approach provides greater flexibility than *direct input mapping*, where the region is defined and displayed in physical space, such as where a coincident floor located display and sensor create something like a touchscreen.

To develop a new interaction technique we needed design guidelines based on empirical evidence. Previous research had examined indirect directional kicking with one foot while standing [1,14,29]. Previous work had also examined “0D” in-place floor tapping while seated [7], 1D pedal tapping while seated [10,17], and 2D interactive floor tapping emphasizing direct input issues such as perceived input point and occlusion [3]. However, indirect tapping while standing has received little attention.

We undertook two experiments to test human performance using indirect foot pointing using discrete taps and kicks while standing. We first tested foot tapping to determine appropriate target size, direction and distance within a practical configuration of annular targets placed in semi-circular rings. We then extended this work by running a second study in which we added kicking to provide a direct comparison with tapping, and examined the effect of removing the foot cursor to simulate truly eyes-free interaction.

Based on the results of two experiments, we created a concise set of ten design guidelines. For example, we found that left and right feet perform at similar levels; that there is little detectable difference in time across target configurations or directions, but targets with an angular size under 22.5° or radial size under 5 cm should be avoided due to high error rates. There is a small advantage to using tapping compared to kicking for pointing actions, but little practical difference. Removing the foot cursor produces high error rates of 27%, but there was room for improving this technique from the one tested in our study.

While our study was conducted with our specific application scenario in mind, this style of foot interaction while standing has broad applicability to many situations when there is reason to avoid, reduce, or augment hand input. In mobile settings, foot input is useful with a head mounted display (**Figure 1-3**a) or when hands are occupied and smartphone input is burdensome [7] (**Figure 1-3**c). In large display settings, foot input could augment finger touches by triggering commands such as “undo” (**Figure 1-3**b). Our design guidelines and interaction technique will also help inform design for these applications.

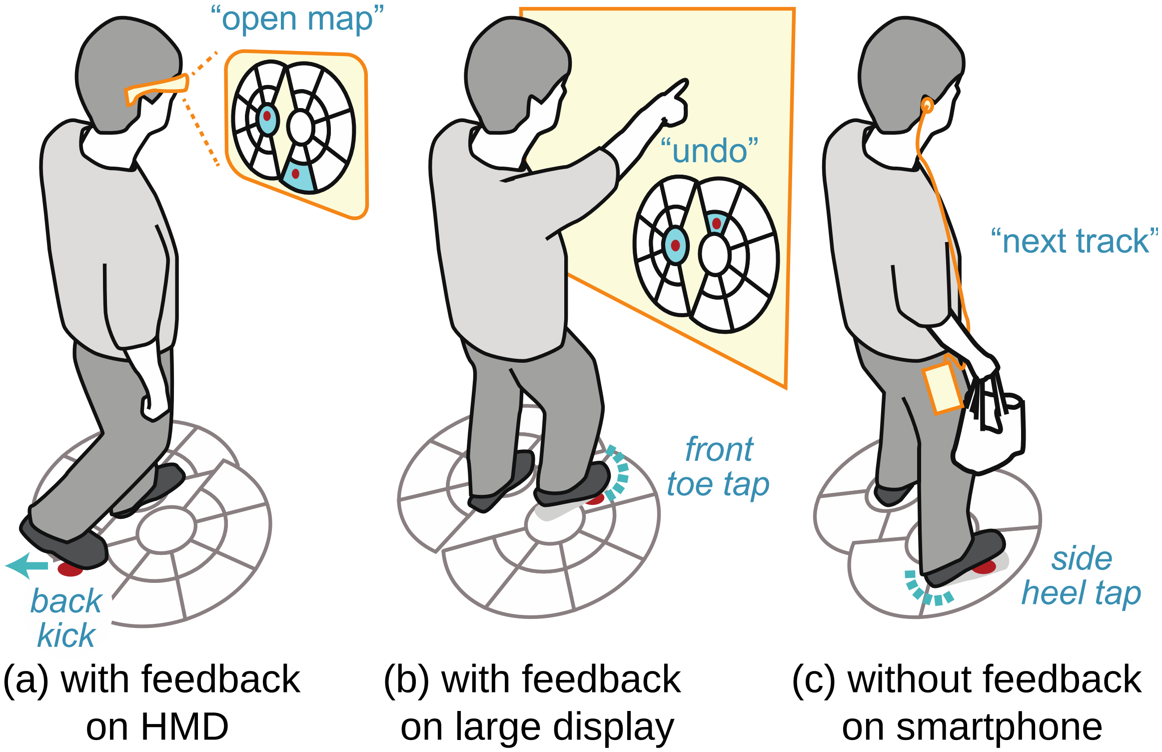


Figure 1-3. Indirect foot pointing
Indirect foot pointing using discrete taps and kicks on virtual targets in semi-circular rings around feet, device and target examples: (a, b) indirect feedback using foot cursor (red dot) on high-density targets with head mounted display or large display; (c) indirect without feedback using low-density targets with smartphone.

1.3 Contributions

In this thesis, we present the following contributions to foot interaction research:

- A set of design guidelines for indirect foot interaction covering a comparison between tapping and kicking; the effect of foot dominance, target size, location and visibility of foot cursor on performance; appropriate interaction ranges; and characteristics of directional taps and kicks.
- A set of indirect, discrete foot interaction techniques for standing desks.
- A method of mapping from foot interaction techniques to application functions that can augment mouse and keyboard input, provide a high level of application control on its own, and discourage cyberslacking. We also include descriptions of how our system can be used for general web

browsing, web-based academic research, and interactive program debugging sessions without modifying existing desktop applications.

- Feedback visualizations that make foot input practical and learnable with real desktop applications. A constantly available side panel displays a foot cursor with virtual target positions and a dynamic cue card showing foot action to command mappings. A help overlay can be summoned to show foot input actions in the context of a GUI application's equivalent commands.
- A subtle interface adjustment and input technique enables a special "click mode" where arbitrary GUI targets are decorated to show a unique sequence of foot actions for selection.
- An approach for foot tracking using an under-desk IR depth-camera and IR LED along with insole-mounted pressure sensors.

1.4 Organization

The thesis is organized as follows:

Chapter 2 describes previous work in the area of foot input, including the design of previous foot input systems and their application scenarios. It also summarizes the relevant previous empirical research which has been carried out on human performance with foot input systems.

Chapter 3 describes two quantitative experiments that we ran to provide design guidelines for an indirect foot input system. It also outlines the basic interaction technique we use in the experiments and in the system.

Chapter 4 describes the design of the Tap-Kick-Click system in detail, including the finalized interaction technique, sensing system, and scenarios describing the system's application to control common web applications. It also includes the results of a user evaluation of the system.

Chapter 5 summarizes the contributions of the previous chapters.

Chapter 2

Related Work

People use their feet to provide input in many situations in everyday life, for example: braking and accelerating a motorized vehicle, steering an airplane, controlling the speed of a sewing machine, and choosing electric guitar effects. As early as 1915, a patent was granted for a foot pedal used to enter carriage returns on early typewriters [43]. Thus, it is not surprising that foot input systems have been proposed for controlling computers. When Doug Englebart's lab was searching for a cursor control input device, they experimented with foot input since it freed both hands for typing [48]. But such attempts proved unsuccessful, and today a simple footswitch for audio transcription is the only mainstream example of feet controlling conventional desktop applications [32]. However, foot-based interfaces have since been applied to, or proposed for, many additional applications such as gaming [18,29], hands-free interaction for mobile devices [1,7,14,39], and large floor or wall displays [3,20].

In this chapter, we survey these proposed and realized applications for foot input, the design of previous systems for foot input, and empirical studies of human performance using foot input. We explain why previously designed systems are not suitable for our intended purpose, and where our input system stands in relation to previous empirical work.

2.1 Applications

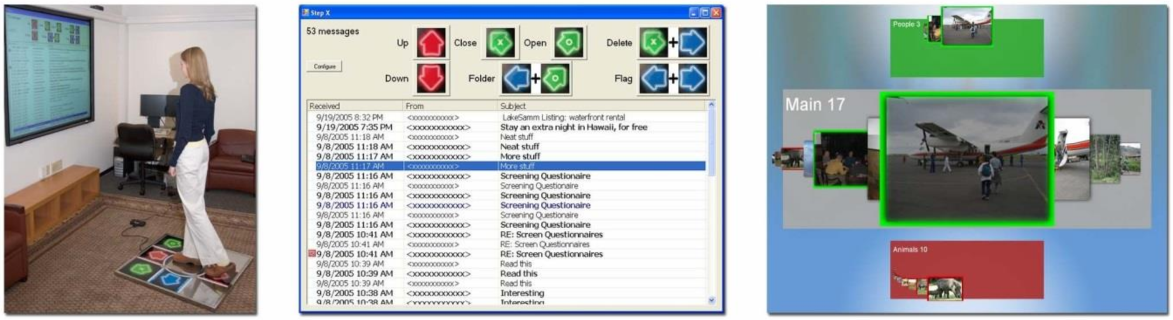
When one thinks of evaluating an input system, the first goal that comes to mind is comparing it with the currently dominant input system (keyboard and mouse for computer control, touchscreens for mobile input) and looking for improvement in general performance across a wide variety of tasks. However, efforts to use feet to increase desktop computing performance have generally been unsuccessful. Kim and Kaber were unable to show a clear benefit for using multiple foot pedals for font style selection in a text editor [21]. Pearson and Weiser found their rate control "foot joystick" to be slower and more error-prone than a mouse [33]. Pakkanen and Raisamo report a foot-controlled trackball is slower, more error prone, and less preferred than a hand-controlled trackball [30]. Dearman et al. [8] used two foot pedals for seated heel and toe tapping to simulate a multimodal text entry technique for mobile devices, and found that although it was comparable with touch input in speed, users made more errors while using foot input and preferred touch to foot input overall. Göbel et al., 2013 [12] propose a design to combine gaze input and foot pedal input for navigation of zoomable information spaces, but do not evaluate their design.

These are discouraging results if the goal is raw performance, but foot input can be used to achieve a variety of other goals that make it useful for specific applications. Foot movement can be used in games, making interaction more immersive and life-like and increasing enjoyment from physical activity [9]. A well-known example is Konami's Dance Dance Revolution (DDR) [18] game, where the player has to make specific foot movements in time with the beat of a piece of music. More generally foot movement is an important part of exergaming [42] and systems such as the Microsoft Xbox Kinect, the Nintendo Wii, and mobile games [29] use taps, jumps, and kicks. Foot input can also be used be used to provide a break from ordinary keyboard and mouse computer use, as proposed by Berque et al. [4] who used their F.U.T. mouse to control a simple game and a scrolling Twitter feed.

Foot input has been proposed for several applications in mobile interaction. It can provide a subtle, socially acceptable alternative to arm gestures; in a study of the social acceptability of mobile gestures, Rico and Brewster [35] found 88% of participants deemed foot tapping to be a socially acceptable input method. Feet can be used to control a phone without removing it from a pocket; Crossan et al. [7] used sequential in-place toe tapping to choose items from an eyes-free menu selection, and found that foot tapping was faster for selections requiring 4 or fewer taps. They can also be used in situations where the hands or a touchscreen is not available, such as when the hands are dirty as suggested by Alexander et al. [1] or when interacting with a head mounted display.

Foot input may also be applicable to providing interfaces for people with certain kinds of disabilities. Pedrosa and Pimentel [34] implemented a text entry system (SwingingFoot) and an interaction technique (DuoGrapher) for a person with a severe motor disability, detecting heel rotations while lying down using a smartphone accelerometer.

Foot input has been used to provide the primary input channel or to augment touchscreen input for immersive environments, where including a keyboard and mouse would disrupt the interaction. Schöning et al. [38] sensed subtle shifts in foot balance with a Wii balance board to augment multi-touch map navigation on a large display. Jalaliniya et al. [19] produced a set of combination hand and foot gestures for use in a surgical setting to interact with medical images without touching a physical interface and compromising the sterility of the environment. In their Multitoe system, Augsten et al. [3] use feet for direct manipulation on an interactive floor, with the aim of allowing much larger interaction areas with more interactive objects than is possible with digital tabletops. Matthies et al. [25] produced their ShoeSoleSense system to allow users to navigate through immersive virtual reality environments without having to use arm gestures, and Carrozzino et al., [5] used an Arduino based pressure sensitive mat for the same purpose.



Received	From	Subject
9/19/2005 8:32 PM	<xxxxxxxxxxx>	LakeSamm Listing: waterfront rental
9/19/2005 7:35 PM	<xxxxxxxxxxxx>	Stay an extra night in Hawaii, for free
9/8/2005 11:18 AM	<xxxxxxxxxxx>	Neat stuff
9/8/2005 11:18 AM	<xxxxxxxxxxxx>	Neat stuff
9/8/2005 11:17 AM	<xxxxxxxxxxxx>	More stuff
9/8/2005 11:17 AM	<xxxxxxxxxxx>	More stuff
9/8/2005 11:16 AM	<xxxxxxxxxxxx>	Screening Questionaire
9/8/2005 11:16 AM	<xxxxxxxxxxx>	Screening Questionaire
9/8/2005 11:16 AM	<xxxxxxxxxxxx>	Screening Questionaire
9/8/2005 11:16 AM	<xxxxxxxxxxxx>	Screening Questionaire
9/8/2005 11:16 AM	<xxxxxxxxxxx>	Screening Questionaire
9/8/2005 11:16 AM	<xxxxxxxxxxxx>	Screening Questionaire
9/8/2005 10:41 AM	<xxxxxxxxxxxx>	RE: Screen Questionnaires
9/8/2005 10:41 AM	<xxxxxxxxxxx>	RE: Screen Questionnaires
9/8/2005 10:41 AM	<xxxxxxxxxxxx>	RE: Screen Questionnaires
9/8/2005 10:39 AM	<xxxxxxxxxxx>	Read this
9/8/2005 10:39 AM	<xxxxxxxxxxxx>	Read this
9/8/2005 10:39 AM	<xxxxxxxxxxx>	Read this
9/8/2005 10:38 AM	<xxxxxxxxxxxx>	Interesting
9/8/2005 10:38 AM	<xxxxxxxxxxxx>	Interesting

Figure 1. Lab Setup
Figure 2. StepMail
Figure 3. StepPhoto

Figure 2-1: Figures 1, 2 and 3 from Meyers et al. [27], showing StepMail and StepPhoto

Closest to the purpose of our research, previous work has shown that it is possible to use physically active game-like foot movements to perform some limited conventional computing tasks. This can providing a break from keyboard and mouse work, but comes at the cost of reduced efficiency. Meyers et al.'s alt.chi paper [27] used 7 tap-sensitive sections of a 3×3 foot switch mat (a standard DDR game mat) to make sorting email and photos more physically active and enjoyable in their StepMail and StepPhoto applications (Figure 2-1). They found a statistically significant increase in participants' heart rates when using the system vs. their resting heart rate, and participant feedback was somewhat positive. They also found that users wanted command mappings to require the minimal number of actions needed for accomplishing the task, even though the goal of using the system was increasing physical activity. They used a limited discrete input vocabulary that was adequate for their simplified applications, but would not scale to more diverse applications with more complex tasks while maintaining reasonable efficiency. An unpublished report by Wu [44] also probes the idea of using a foot-operated mouse at a standing desk to promote "healthy movements," but the underwhelming results for seated, foot-based cursor control suggest that a foot mouse will be needlessly inefficient and does not provide physically active game-like movements.

In our research, we refine these ideas to apply foot based movement to perform a more general class of interaction tasks, with the intent of providing a healthy and fun break from seated work with the keyboard and mouse. No published work has combined foot input with a standing desk, and there are no examples attempting command-level control of conventional desktop applications.

2.2 Design Choices

2.2.1 Sensing

Previous studies have used a variety of sensing techniques to track foot movement for the purpose of interaction. Studies looking at foot input for desktop applications have used a variety of physical input devices, such as foot pedals [21] or trackballs [30]. Camera based sensing has been used to detect foot movement for kick gesture recognition, using the Microsoft Kinect depth camera [1,14] or custom computer vision algorithms [29]. For example Lv et al. [23] implemented a tracking algorithm that uses a smartphone camera to detect kick movements, even when the camera is used in unusual poses. Floor based sensors have included pressure sensitive mats [5,27] and capacitive sensors [19]. An elaborate example of a floor based sensing system is the Multitoe system, where the floor is replaced with a projection screen and Frustrated Total Internal Reflection (FTIR) is used to detect pressure applied on the floor.

Pressure and inertial movement sensors in shoes and socks have been applied to fitness tracking in commercial products such as the Footlogger insole [49] and the Sensoria Smart Sock [50], but wearable sensing can also enable interaction. Accelerometer sensors have been applied to detection of tapping [7,39] and other foot gestures [45]. Paradiso and Hu [31] used sensors in shoes for an interactive dance performance and Matthies et al. [25] used insoles with pressure sensors to detect in-place actions like jumping, walking, turning, and toe movements. Studies like Dearman et al. [8] used physical foot switches, but they intended their study to simulate a mobile interaction technique using on-foot sensing.

Feedback is usually provided to the user via a display, but also can be provided via a vibration motor attached to the foot or shoe [23]. Tangible feedback can be provided by having users interact with objects on top of a pressure sensitive floor (“Kickables”) [37], as implemented by Schmidt et al.

In our initial study we prototyped the interaction technique using a camera-based sensing system, the Vicon motion capture system, for ease of setup and flexibility of prototyping. For our final system we elected to combine camera based sensing using the Kinect to reliably detect absolute position of the foot, and insole pressure sensors to make tapping more expressive and robustly detectable.

2.2.2 Foot Actions and Gestures

Previous studies have used two different kinds of foot actions to express information to the computer. *Continuous* foot actions provide information on some numerical scale, such as measuring the velocity of a kick, and can be mapped to continuous control of position or speed of movement of a cursor or object on a display. In contrast, systems using *discrete* foot actions take a set of distinguishable foot movements and map them on to separate commands, such as using taps on specific locations on the floor to type letters. Alexander et al. [1] included both types of commands. The paper included an elicitation study which had participants assign discrete gestures to various types of commands for a mobile device, and then the authors implemented a gesture recognizer using an accelerometer to distinguish between these gestures. The authors then investigated continuous gestures for map navigation. They used kick based gestures with a tapping delimiter and mapped displacement or velocity of the kick to displacement or velocity of the map.

While tapping and kicking are frequently used in foot input systems, other gestures have been proposed and investigated. For example, Scott et al. [39] explored single foot gestures including dorsiflexion (raising the toe with the heel on the ground), plantar flexion (raising the heel with the toe on the ground), heel rotation (rotating the foot with the heel planted) and toe rotation (rotating the heel with the toe on the floor). They found better performance with plantar flexion vs. dorsiflexion, and that users preferred heel rotations over toe rotations. They also implemented a classifier capable of using accelerometer data from a phone in the pocket to sense these actions, but their system required an in-place toe tap to demarcate interaction.

In our performance study and implemented system, we use a combination of kicking and tapping with movement, so that the user taps on or kicks over virtual target locations on the floor. This provides moderate physical movement, is easily sensed, and allows for a large number of commands to be expressed in a simple manner. We also include a special gesture, a two foot jump, to access the help menu in the complete system.

2.2.3 Direct and Indirect Input

Meyers et al. [27] used a mat on the floor divided into a 3 by 3 grid of pressure switches to provide foot input. In their Multitoe system, Augsten et al. [3] had users stand on a projection screen and used a camera based method to sense pressure of the foot on the floor, creating a system on the floor analogous to a touchscreen. Both of these systems represent *direct foot pointing*, where interaction takes place when the foot is tapped on or moved over a target in physical space indicated using a visual cue, such that the user looks at the floor to determine the location of the target. Direct foot pointing creates a “fat foot” problem, where the foot covers multiple target locations and it is ambiguous which one was intended to be interacted with. Augsten et al. found that the mental model for the selection “hotspot” (the point on a foot used select a target) varies by individual from toe, offset from toe, and ball of foot. They solve this by allowing each user to select their own hotspot location on the foot. Direct foot pointing also can create issues with occlusion when information is present on the floor as in Multitoe. If information is present on a separate display users may need to split their attention between the display and the floor while learning the system or to troubleshoot errors.

The alternative to direct foot pointing, which we used in our system, is *indirect foot pointing*. In indirect foot pointing a hotspot is selected on the foot, foot position is represented by a cursor, and both the target location and cursor are displayed on a screen. This model allows the user to direct their attention at a location other than their feet, making interaction more comfortable and less prone to occlusion. It allows reconfigurable pointing interaction without requiring expensive hardware capable of displaying and sensing pressure on the floor.

2.3 Empirical Studies

We briefly survey studies relevant to our system, looking both at direct and indirect foot input.

2.3.1 Foot input While Seated

Most of the studies of foot input while seated are not appropriate to develop design guidelines for standing input given, differences in balance and range-of motion. Controlled studies by Drury [10] and Hoffmann [17] have confirmed that foot motion follows Fitts's law for 1D foot movement along a line while seated, and is slower than comparable arm movements. However, the characteristics of 2D foot movement while standing are sufficiently different, involving weight shifting and different motion restrictions, that we do not have confidence that this result generalizes. Garcia and Vu [11], found that foot input using a foot mouse was harder to use than a hand trackball (chosen to be unfamiliar to the participants for a fair comparison for foot input) even with multiple sessions of practice, but they found that users improved foot input performance more than they improved hand input performance over the training period. This suggests that part of the increased difficulty of foot input may be a lack of previous exposure to any foot input, and a greater period of time may be needed for training to observe true performance levels.

2.3.2 Foot Input While Standing

Augsten et al. [3] investigate foot input on the Multitoe interactive floor display. Their findings related to indirect input performance include minimum target sizes for a foot keyboard and selection of a "hotspot" (the point on a foot used to select a target). They found that 3.1 cm by 3.5 cm targets were needed for a reasonable (10%) error rate, while 5.3 cm by 5.8 cm targets achieved a low error rate of 3%. They found the perceived hotspot varies by individual from toe, offset from toe, and ball of foot. These results have some relevance, but our work uses an indirect input style. Additionally, differences between direct and indirect touch input [36] are likely to translate to feet.

The most relevant tapping study is the Meyers et al. [27] StepMail and StepPhoto applications using a DDR game mat. No controlled experiment was conducted, but a usability evaluation found people wanted the mapping from commands to buttons to facilitate alternating or balancing taps between their two feet. Participants also enjoyed commands that required them to jump (which deleted emails in one of their applications), but didn't want the command mapping to require more commands to perform a task than was strictly necessary. The Meyers et al. system serves to validate the discrete, indirect foot pointing input space we investigate.

Although there is little previous work investigating tapping (aside from Crossan et al. [7] which found an in-place “0D” tap took 1.2s), there has been considerable interest in kicking. Han et al. [14] examined the direction and velocity characteristics of forward kicks. They found that people could reliably kick in 5 distinct forward directions over a 120° arc (24° targets) and produce two distinguishable levels of kick velocity. However, it is not clear whether participants looked at their feet or only at the tablet display during the task. Alexander et al.'s [1] elicitation study suggests people prefer spatial taps and kicks for certain tasks. They explore single-foot kick characteristics for controlling continuous map navigation and provide basic guidelines: backwards kicks are difficult and controlling kick direction is easier than kick distance. They do not investigate tapping performance beyond using an in-place foot tap like Crossan et al. [7] to stop navigation. Neither Han et al. or Alexander et al. evaluate spatial kicking or tapping for discrete target selection.

Overall, these studies present tentative suggestions for target sizing and preference. Augsten et al. [3] suggests targets of between 3cm to 6cm will achieve reasonable error rates for direct tapping. Han et al. [14] suggest targets of radial size 24° for kicking. The Meyers et al. study used a DDR mat consisting of a 3 by 3 grid of footswitches 86cm (34 inches) on a side, forming targets of approximately 30 cm for eyes-free tapping. However they did not report time or error rate information, or vary the target size to determine if this is optimal. Alexander et al. [1] also suggest that direction matters, as backwards kicks were more difficult.

Although these studies provide a useful starting point, none of them compare tapping and kicking directly, or evaluate cursor controlled indirect input. In Chapter 3, we address these gaps in the previous work by describing the results of two quantitative experiments on the performance characteristics of foot input.

Chapter 3

Performance Characteristics of Foot Input

Before we could design an interaction technique using foot-based input to control conventional applications at a standing desk, it was essential to understand several fundamental performance characteristics of human movement. We found some foundational insights in previously published work, but we needed answers to the following questions before we could proceed:

1. Do users prefer or perform better when using a tapping action or a kicking action?
2. Do users perform better when using their dominant foot than using their non-dominant foot?
3. Do users have a preference for which pointing action to use in discrete target selection?
4. Does user performance vary by the direction of movement (moving the leg forwards, out to the side, or backwards)?
5. How large do targets need to be in order for users to be able accurately select them?
6. How does the type of feedback provided by the system impact user performance?
7. How does the distance the foot has to move impact user performance?
8. What size of area is appropriate for foot interaction?

To answer these questions in the context of designing a novel foot interaction technique, we ran two closely related quantitative experiments to investigate human performance in a controlled setting, supplemented with qualitative interviews, questionnaires, and observations. Two experiments allowed us to investigate different issues without extending the time required of the participant, and allowed refinement of the second study based on the results of the first. The goal of our first experiment was to investigate discrete, indirect foot pointing using taps on a range of radial and angular target sizes. Once we established a usable range of target sizes, our second experiment compared pointing using taps to pointing using kicks, and tested the limits of eyes-free indirect input with a no cursor condition. The studies yielded a rich dataset which answered the questions we initially considered, as well as providing additional insights.

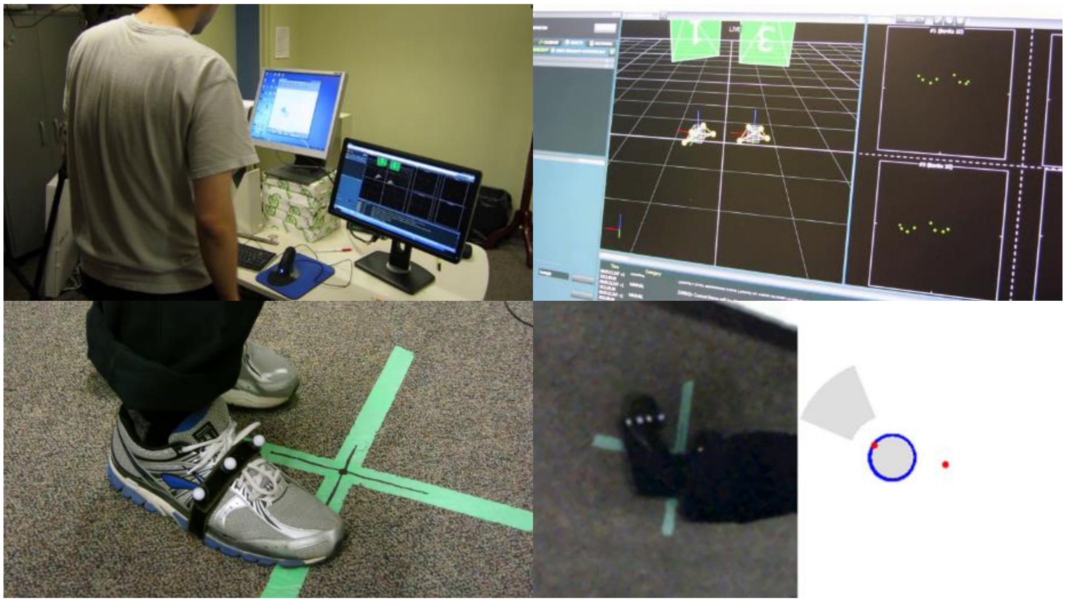


Figure 3-1: Video Figure for Chapter 3
This chapter is accompanied by a video figure, which accompanied our submission to GI 2015 to demonstrate the experimental setup and interaction technique described in this chapter. The video is available at:
http://williamsaunders.net/thesis2015chapter3
or http://youtu.be/mCI5fXKXrpo

3.1 Interaction Technique

When formulating our research questions, we had a tentative interaction technique in mind. This interaction technique fulfilled our purpose of supporting use of desktop applications in a standing desk environment, and was not previously explored in the literature. It assumes indirect input, where a hotspot is defined on the foot and used to provide a single point for the foot's location, like a cursor on a mouse corresponds to a specific physical location on the device. This technique also primarily uses discrete foot input instead of continuous input. We were interested in command invocation as a primary goal for the system, which maps naturally onto discrete input. Discrete input is also easy to define, explain and sense, especially compared to some proposed continuous interaction techniques such as using kick displacement for rate controlled navigation [1]. The technique has some conceptual similarity to the DDR mat used in Meyers et al. [27], but provides the benefits of indirect input such as greater flexibility, and provides a greater number of commands invokable with a single action.

The technique is divided into 6 steps as follows:

1. The user stands normally with both their feet a comfortable distance apart in a “home” position. While a foot is close to its home position, the system does not trigger any commands. This leaves a simple and easy position for the user to assume when they don’t want to interact, such as when reading text.
2. The user lifts up their foot from the ground, and moves the foot away from the home position in some direction.
3. The user positions their foot such that it is over a target, which is a specific region in indirect input space.
4. While over the target, the user performs some discrete, detectable pointing action. This could include tapping some part of the foot on the ground, hovering, or simply reversing the direction of motion of the foot.
5. The system senses the combination of target and pointing action, and performs the corresponding command (some behavior, such as an application command, emulating a keystroke, etc.). The correspondence between the pair **target** + **pointing action** and the command is defined in a *command mapping* known by the user.
6. The user moves their foot back to the home position, while keeping it in the air and avoiding performing a pointing action over any other target (to avoid unintentional activation of another command).

While the user is performing this sequence, some method of feedback is used to inform the user about the state of the system and help them to troubleshoot errors. It can be provided by using a foot cursor on a display the user is looking at, but other methods of feedback may be compatible with the technique.

An important consequence of a design of a system implementing this interaction technique is its command throughput, which is the number of commands that can be sent in a given amount of time, analogous to the characters per minute measurement of typing speed. Increasing the number of sensed pointing actions and targets will increase the number of commands that can be performed in one movement cycle. However, this may require decreasing target size and moving targets further away from the home position, which could increase movement time and increase error rate (which would require the user to perform additional actions to correct the misrecognition). While command throughput is unlikely to reach the level achievable in a traditional keyboard and mouse interface, greater throughput may make the difference between a useable system and one that is too difficult to control.

3.1.1 Design Decisions Prior to Experiment

Within this framework, we made additional design decisions prior to running the experiments:

While a real system might allow the user to perform multiple **target** + **pointing action** combinations before returning to the home position, we limit the user to one such combination in this experiment. The simplicity of this action sequence reduced the complexity of the study, and allowed for simple detection of kick actions in a similar manner as tap actions.

We chose to fix the position of the foot hotspot in the center of the foot rather than allowing for a user selected hotspot. While Augsten et al. [3] found that perceived hotspot varies by individual for direct input, we do not think this result carries over to our interaction technique. First, choosing the intuitive hotspot is more important in direct input, where the user is looking at the target and using visual control to guide their foot over the target location). This interaction technique uses indirect input, where the user does not need to look at their feet, giving us the freedom to define the mapping between foot motion and the motion of an implicit or explicit cursor. Second, we were interested in discovering what part of their foot participants would use to tap with when not prompted. If users could have selected hotspots in different parts of the foot, this would have biased the choice of foot action - a heel hotspot would shift as the pitch of the foot changes when the user tries to tap with their toe, where a toe hotspot would not. By selecting the midpoint between heel and toe positions as the hotspot, we avoided biasing choice of foot action in experiment 1.

We chose to fix target shape to pieces of a two-dimensional ring, called “annular sectors” (Figure 3-2a). This choice of shape arises naturally from representing foot movement to and from the home position as an action in radial coordinates and assuming that foot movement time and difficulty are mainly controlled by the distance from the center the foot has to move, which has been done in previous studies such as Han et al. [14]. This seems to be the most biomechanically natural way to parameterize foot movement in this interaction technique, as foot movement is produced by rotating the leg while the user stands on one foot with their hip in a fixed position. Note that circular or rectangular targets, which would be used in a performance study of mouse movement, would present an irregular profile to the user in radial width and radial distance. Annular sectors are also appropriate for the interaction technique in that they represent the highest density packing of a multi-layered ring of targets located at a constant distance (Figure 3-2b), whereas spherical or rectangular targets would have irregular gaps between them.

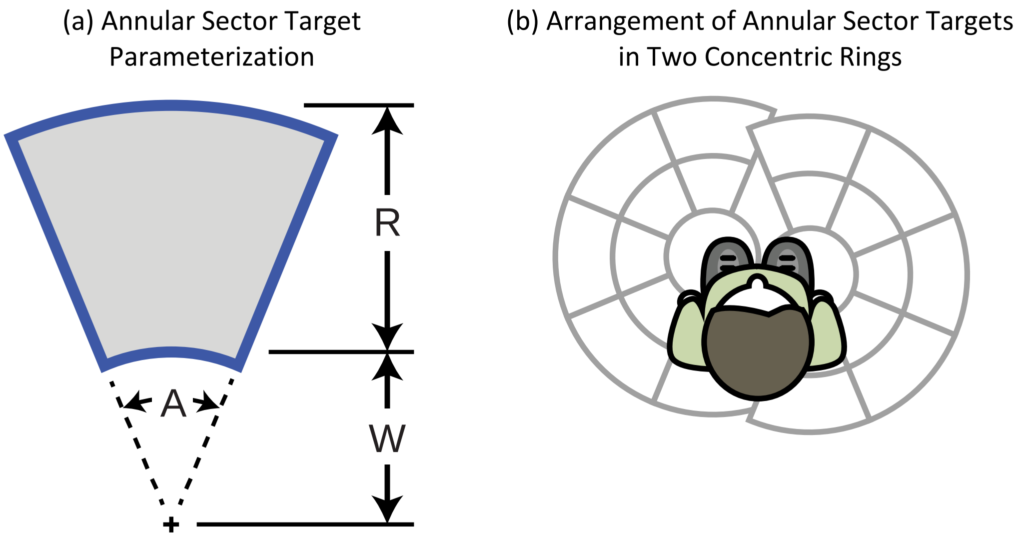


Figure 3-2: Targets used in investigated design technique
a) Annular sector shaped targets used in the experiment, and parameterization of target size and distance; b) Arrangement of annular sector shaped targets into two concentric rings for foot input

The choice of annular sectors produces the following parameterization of target size: angular size A, radial width R, and distance to inner edge of the sector W (Figure 3-2a). The direction of motion is then represented by an angle relative to the forward direction. This allows for a complete representation of a multi-layered ring. Distance to the inner edge was selected as participants could interact anywhere within the target, and we assumed prior to the study that they would likely choose to interact closer to the inner edge for large target sizes and distances.

We chose to provide feedback to the user only through the display, rather than in the physical environment, as this allowed greater flexibility in redefining the target layout and did not require the user to split their visual attention between multiple locations.

3.2 Experiment 1: Tapping

The goal of this first experiment was to investigate discrete, indirect foot pointing using taps on a range of radial and angular target sizes. We determined which target sizes would work well, with reasonable time and error rate, and then applied this knowledge to the second experiment.

3.2.1 Participants

Eleven people from a university campus (3 female), ranging in age from 20 to 37, participated. 11 reported they were right-footed (i.e. they kick a ball with their right foot) and 3 reported they had previously used a whole body input device. Participants were screened to exclude anyone with an injury or impairment that would interfere with their performance or lead to further injury. 13 people were originally recruited, but two were excluded prior to quantitative analysis: one had unusually high tracking errors and one used an unanticipated strategy to complete the tasks involving sliding the foot along the ground instead of breaking contact with the floor and lifting it up. While this strategy allowed the participant to perform the task, it might cause problems in a real system (where other targets might need to be avoided). Data for this participant was discarded prior to analysis to avoid clouding the results by combining two separate strategies for completing the task.

3.2.2 Apparatus

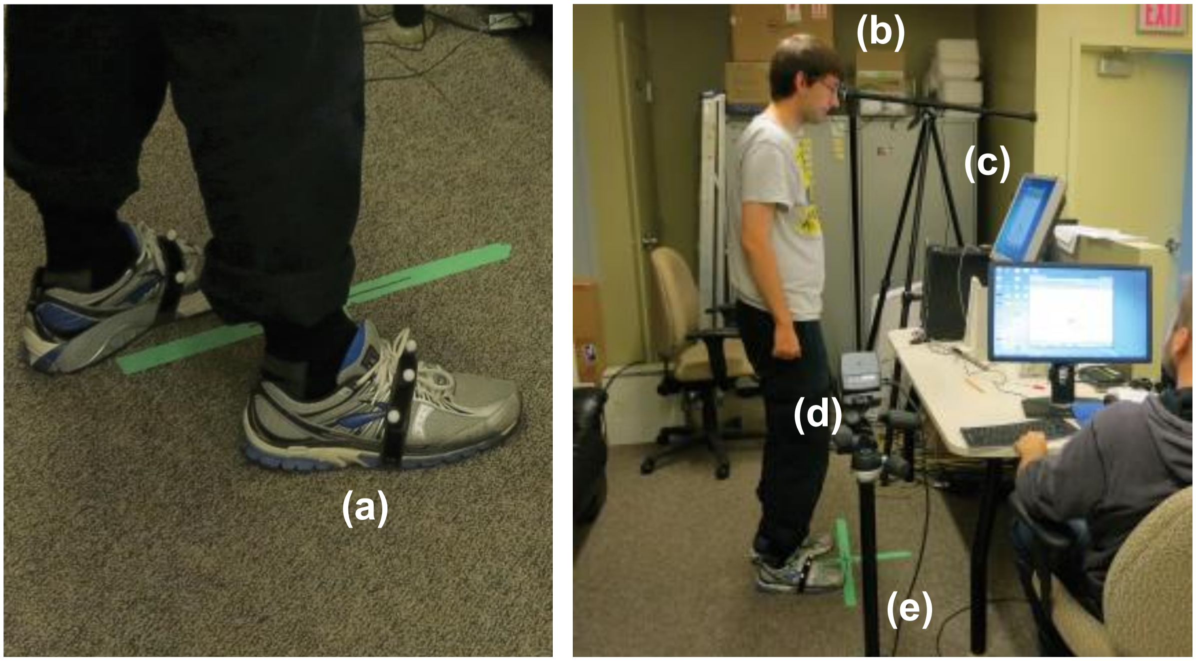


Figure 3-3: Apparatus
(a) Vicon tracking markers attached to both shoes; (b) video camera to record session; (c) desktop monitor on raised desk platform for experiment feedback; (d) Vicon tracking cameras; (e) marked calibration position on floor.

In the quantitative experiments, our aim was to measure performance under ideal tracking conditions, establish an upper bound on performance possible from less obtrusive tracking systems such as the Microsoft Kinect, and clarify design principles before spending time developing additional sensing hardware and algorithms. For this reason we used a Vicon motion tracking system for high-resolution, high frame rate (100 Hz), low latency data. The 3D position and orientation of both feet were tracked using infrared reflective markers on elastic bands wrapped around each foot (Figure 3-3a).

A 17-inch display on a raised stand in front of the interaction area displayed all experiment visuals (Figure 3-3b). All visuals were easily legible from 0.75 m away, the typical distance from participant to display. In addition to logging movement to Vicon capture files and logging all input events in our software, we also video recorded sessions for qualitative analysis (Figure 3-3a).

3.2.2.1 Foot Cursor Hotspot Calibration

We calibrated for each participant’s shoe size by recording offsets from the tracked position of the band to the heel and toe by getting participants to tap their toe and heel on a floor registration point

(Figure 3-3e). We then took the midpoint of the toe and heel position as the center of the foot, which was used as the hotspot. Visual feedback in the form of two red circles (“foot cursors”) represented the real-time hotspot position for each foot. The foot position in motor space was mapped to display space using a constant CD Gain of 8.5 px/cm.

3.2.2.2 Target Selection Action Detection

Selection of targets (like mouse “click” events) was triggered using thresholds determined in a small number of pilot experiments. When the height of either the toe or heel transitioned below 4 mm above the floor and momentary foot speed was less than 0.2 m/s, a selection event was triggered. To avoid hysteresis, the foot had to lift more than 8 mm above the floor or travel at a speed greater than 0.3 m/s before a previous selection event was cancelled. The speed threshold reduced false positives due to uncertainty from deformation of the shoe, but did not reduce the possibility to make rapid taps. These features made selections feel like tapping the floor, and allowed for tapping with the toe, heel, or whole foot with equal ease.

3.2.3 Tasks and Stimuli

To complete each discrete foot-tapping task, participants performed the sequence of actions described in section 3.1 (Interaction Technique). They lifted their foot off of the home target, moved it in the air until the foot cursor was over the task target, and tapped the floor. Then, they immediately returned their foot to the home position by lifting, moving, and tapping on the home target. This rapid cycle was repeated 3 times in succession for the same foot and task target, in order to obtain a more accurate estimate of time and error rate. Images from the system are used to illustrate the interaction technique in Table 3-1

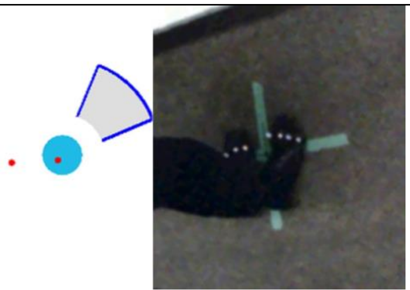
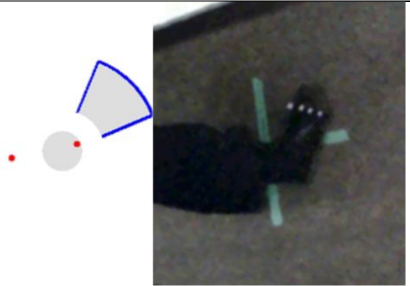
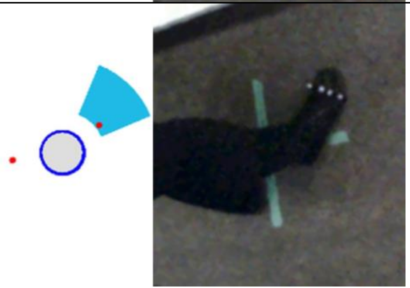
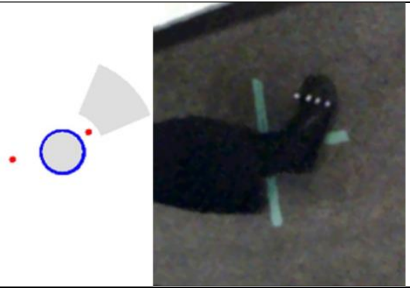
1. The user stands with feet in a “home” position.	
2. The user lifts up their foot from the ground, and moves the foot away from the home position towards the target.	
3. The user positions their foot such over the target.	
4. While over the target, the user performs some discrete, detectable pointing action, in this case tapping on the ground. 5. The system senses the combination of target and pointing action and responds.	
6. The user moves their foot back to the home position, while avoiding activating another target.	

Table 3-1: Realization of the interaction technique from section 3.1
In all images, the left side is the feedback display shown to the user, the right side is an image of the user’s legs and feet on the floor.

3.2.3.1 Target Size and Distance

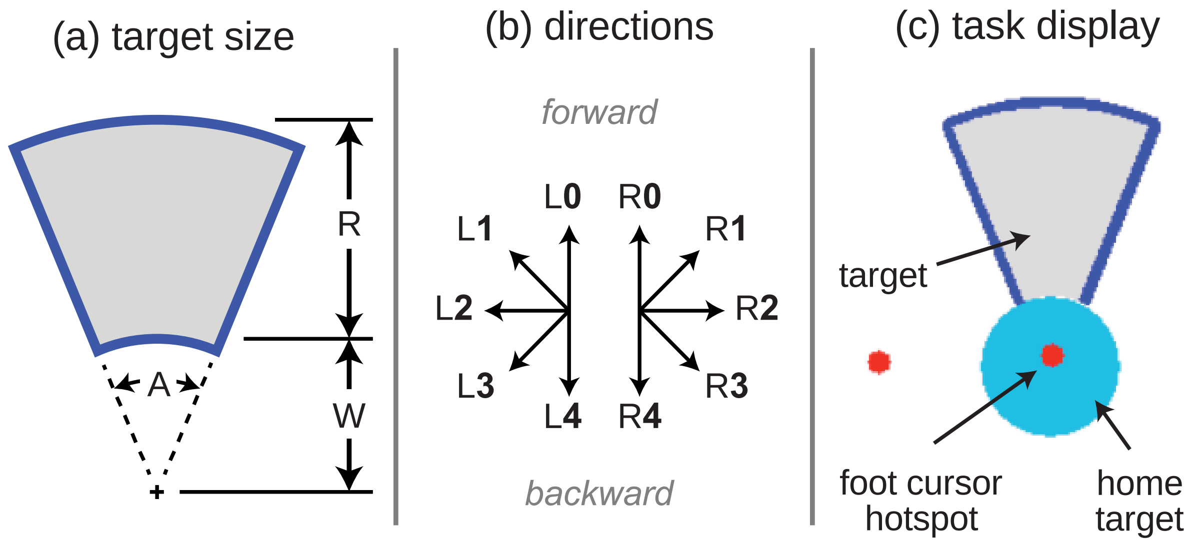


Figure 3-4 Task parameterization and visualization
(a) parameterization of target size and distance;
(b) target directions for each foot, letter indicates foot, number indicates direction;
(c) example task display stimuli for A4R4 (45°, 20cm) R0-forward target at distance 7.5cm, right foot is on home.

Each task was parameterized by 3 variables: angular size A, radial width R, and distance D (Figure 3-4a). All target sizes are actual size in motor space on the floor. Target SIZE is fully defined by the pair (A, R). A is the angular size in degrees and R is the radial width of the target in cm. Informed by small pilot studies, we chose A and R values as whole number multiples of 11.25° for A and 5 cm for R. We use the concise target size notation A*i*R*j* where angular size A is *i* times 11.25° and radial width R is *j* times 5 cm. The set of target SIZES are: A4R4 = (45°, 20 cm), A4R2 = (45°, 10 cm), A4R1 = (45°, 5 cm), A2R4 = (22.5°, 20 cm), A1R4 = (11.25°, 20 cm).

The fixed home position for each foot was calibrated when the participant stood comfortably near the center of the interaction area. The home target represented a non-interactive area where the foot can rest between issuing commands. Two circular home targets represent these positions in the interface; each has a radius of 7.5 cm in the interaction area. Pilot tests determined this size was sufficiently constrained, but allowed the user to reliably return their foot to the home position.

Target distance was measured to the inner edge of the target. We asked participants to tap anywhere inside the target. Assuming they would tap with as little movement as possible, the distance to the inner edge of the target is a more representative distance than target center. We tested two values to investigate the effect of needing to avoid inner targets while interacting with outer target in a system using multiple target rings. A 7.5 cm distance positioned the task target right against the home target and a 15 cm distance created a 7.5 cm gap between the task target and home target like a target on an outer ring. The gap functions as a distractor target which must be avoided.

While this design may seem similar to a Fitts's Law style study (e.g. [10]), our goal was not to compare performance across devices, or produce a fully general model of foot motion. Our focus was to guide interaction technique design similar to the approach of previous studies [14,39]. For this reason, we test different variables, including which foot is being used and direction (along with tapping, kicking, and level of feedback in Experiment 2). In addition, the spatial layout is chosen to match the physiology of leg and hip motion which are not core to traditional Fitts's Law studies where difficulty of motion is more uniform in the interaction space, as with keyboard and mouse.

3.2.3.2 Target Directions

The targets for each foot were positioned at one of five directions (Figure 3-4b) 0 – forward, 1 – forward-diagonal, 2 – side, 3 – backward-diagonal, and 4 – backward. Using the largest angular target size of 45°, the edges of all targets in a real system would touch, making maximal use of the available 5 direction interaction area without overlapping.

3.2.3.3 Target Feedback

At the start of the task, a home target and task target appeared on the side of the display corresponding to the foot required for the task (Figure 3-4c). A purple border around the edge of the target indicated the target to tap next. The target was highlighted in bright blue when the system detected a foot cursor hotspot inside the target region and part of that foot (either the heel or toe) was touching the floor. If an error occurred, defined as tapping while the hotspot was outside of the task target, the purple border moved to the home target (no other feedback was given). Participants had to achieve three error-free repetitions to complete the task. Tasks alternated between feet to reduce fatigue.

3.2.4 Design and Protocol

The independent variables investigated in this experiment are FOOT (left or right), target SIZE, target DIRECTION, and target DISTANCE. Tasks were divided into 10 target configuration sets of 10 tasks. Each set covered all values of 5 target DIRECTIONS and FOOT for one target SIZE and DISTANCE, with a random ordering that always alternated between feet. All target configuration sets were presented in random order as one BLOCK, covering all 300 task settings. Participants completed three BLOCKS in order to test for learning effects.

A short instruction and demonstration block was presented at the beginning and rest breaks were provided at the end of each block. Participants were interviewed after the experiment for subjective feedback about fatigue and preference for toe or heel tapping. The experiment took 60 minutes on average.

In summary the design was:

3 BLOCKS ×
2 FEET × 5 target DIRECTIONS ×
5 target SIZES × 2 target DISTANCES ×
3 repetitions of serial selections
= 900 data points per participant

3.2.5 Analysis

The dependent variables are ERROR RATE, SELECTION TIME, and ROUND TRIP TIME.

ERROR RATE was calculated as the mean percentage of errors per repetition. Errors are defined as when the system detected that the participant tapped their foot while the foot hotspot was not on the desired target. To complete a selection and continue the task, the participant had to successfully tap on the target.

SELECTION TIME is defined as the time duration between the moment the participant had lifted their foot off the home target to the moment when their foot touched down on the task target. SELECTION TIME is averaged over all repetitions in the task.

ROUND TRIP TIME is defined as the mean time to select the task target and return to the home target, including the stationary time at each target. This measurement captures the full time needed to select a target and return the foot to the home position. ROUND TRIP TIME is averaged over the second and third error-free repetitions in each task to avoid any possible effect of the weight shifting as the participant switches feet in the first repetition. Only error-free repetitions are included in time measurements.

3.2.5.1 Outliers

Trials times more than 3 standard deviations from the mean for a target configuration were removed. This removed 162 out of 9900 (1.64%) SELECTION TIME data points and 52 out of 9900 (0.53%) ROUND TRIP TIME data points.

3.2.6 Results

Means over all conditions and participants were: SELECTION TIME: 309ms, ROUND TRIP TIME: 1203ms, ERROR RATE: 14.3%. ROUND TRIP TIME is 4 times longer than SELECTION TIME because it includes stationary time at each target. All tests of main effects use a repeated measures ANOVA and all post hoc tests use a Bonferroni adjustment. A Greenhouse-Geisser correction is applied in the ANOVA where Mauchly's Test of Sphericity is significant, and corrected degrees of freedom are reported.

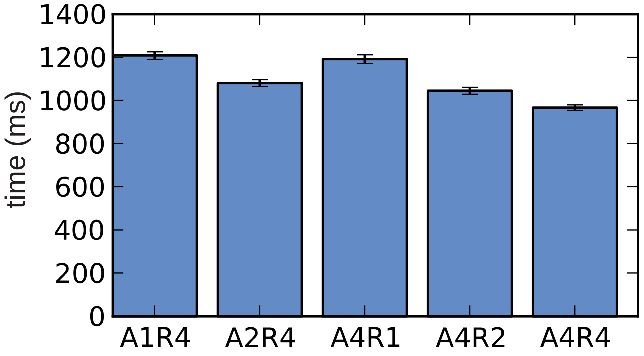
3.2.6.1 Learning Effect

A significant effect was found for BLOCK on ROUND TRIP TIME ($F_{1.074,10.74} = 24.451$, $p < .001$), and SELECTION TIME ($F_{1.077,10.77} = 8.917$, $p < .001$) but not ERROR RATE ($F_{2,20} = 3.073$, $p = 0.069$). Post hoc pairwise comparisons revealed a significant difference between all three blocks for both time measurements. SELECTION TIME decreased by 38ms (11%) from block 1 to 2 and by 19ms (6%) from block 2 to 3. ROUND TRIP TIME decreased by 259ms (18%) from block 1 to 2 and by 109ms (9%) from block 2 to 3. Given the decreasing learning trend, we only discard data in block 1 for the rest of the results.

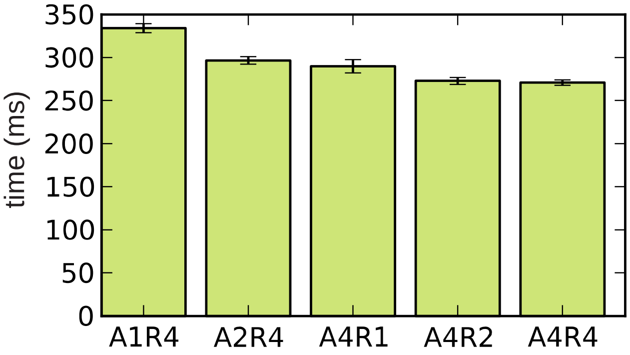
3.2.6.2 Foot

No significant effect was found for FOOT on ERROR RATE ($F_{1,10} = 1.027$, $p = 0.335$), FOOT on ROUND TRIP TIME ($F_{1,10} = 2.816$, $p = 0.124$), or FOOT on SELECTION TIME ($F_{1,10} = 4.854$, $p = 0.052$). 95% confidence intervals indicate that if any difference, it is less than 4.7% for ERROR RATE, 46ms for ROUND TRIP TIME and 35ms for SELECTION TIME.

(a) round trip time by target size



(b) selection time by target size



(e) error rate by target size

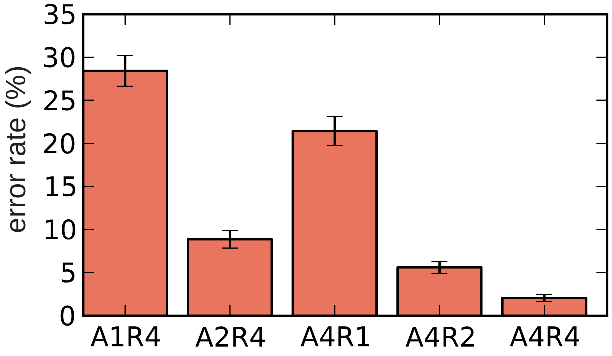


Figure 3-5. Effects of Target SIZE
Effect of target SIZE on: (a) ROUND TRIP TIME; (b) SELECTION TIME; (c) ERROR RATE (all error bars are 95% CI).

3.2.6.3 Target Size

A significant main effect was found for target SIZE on ERROR RATE ($F_{2.249,24.94} = 36.005$, $p < 0.001$). Pairwise comparisons showed that A4R4 had the lowest ERROR RATE of 3% compared to all other sizes, and A4R2 (9%) and A2R4 (6%) had a lower mean ERROR RATES compared to A1R4 (28%) and A4R1 (21%) (Figure 3-5c).

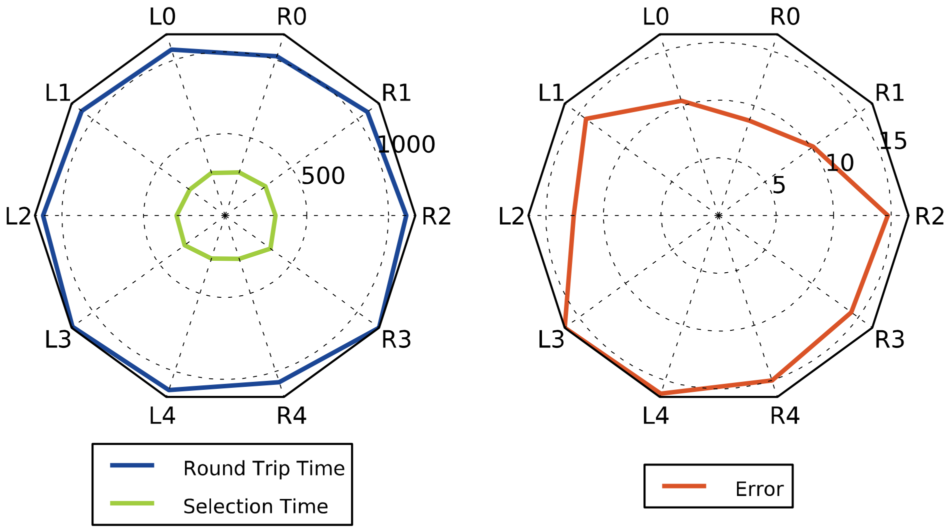
Significant main effects were found for target SIZE on SELECTION TIME ($F_{1.738,17.38} = 6.617$, $p = 0.009$) and target SIZE on ROUND TRIP TIME ($F_{1.930,19.30} = 11.803$, $p < 0.001$). For SELECTION TIME, A4R4, A4R2 and A2R4 had a significantly lower mean value of 280ms, compared to A1R4 at 334MS (Figure 3-5b). The ROUND TRIP TIME of A4R4 was 966ms, significantly lower than A1R4 (1208MS) and A2R4 (1081ms); both A2R4 (1081ms) and A4R2 (1045MS) were also significantly lower than A1R4 (1208MS) (Figure 3-5a).

3.2.6.4 Target Distance

A significant effect was found for target DISTANCE on SELECTION TIME ($F_{1,10} = 64.139$, $p < 0.001$) and target DISTANCE on ROUND TRIP TIME ($F_{1,10} = 49.938$, $p < 0.001$). Pairwise comparisons revealed a difference of 83ms [60ms, 107ms] (95% CI in square brackets) or 25% for SELECTION TIME and a mean difference of 134ms or 11% [92ms, 177ms] for ROUND TRIP TIME (7.5 cm distance lowest for both). There was no effect for DISTANCE on ERROR RATE ($F_{1,10} = 0.105$, $p = 0.75$).

(a) time (ms) by direction

(b) error rate (%) by direction



(c) error rate (%) by direction by task, for distance = 7.5 cm

(d) error rate (%) by direction by task, for distance = 15 cm

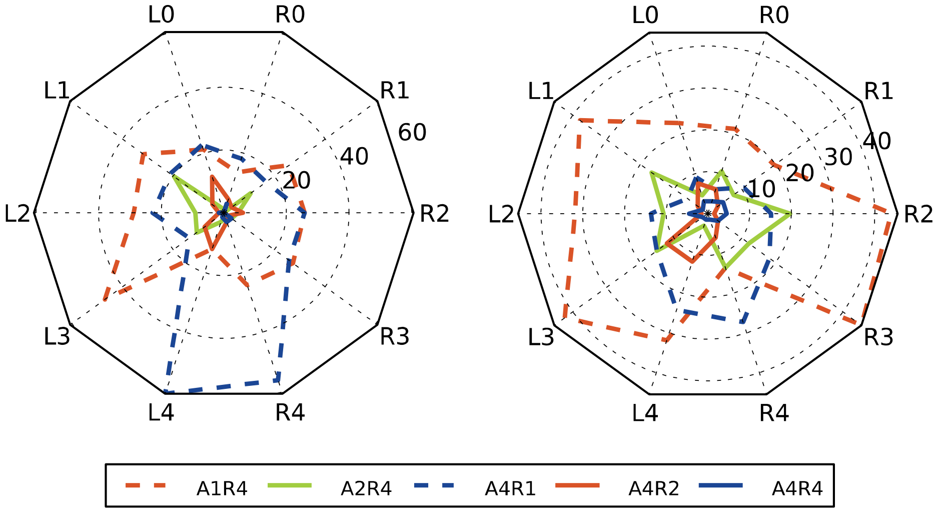


Figure 3-6: Effect of Target SIZE and DIRECTION
(a) time by DIRECTION for all tasks; (b) ERROR RATE by DIRECTION for all tasks;
(c) ERROR RATE by DIRECTION by target SIZE for DISTANCE = 7.5 cm;
(d) ERROR RATE by DIRECTION by TARGET SIZE for DISTANCE = 15 cm.

3.2.6.5 Target Direction

A significant main effect was found for target DIRECTION on SELECTION TIME ($F_{1.689,16.89} = 5.911$, $p = 0.014$) and ROUND TRIP TIME ($F_{1.881,18.81} = 6.384$, $p = 0.009$) (Figure 3-6a). For ROUND TRIP TIME, direction 0-forward was significantly lower than 4-backward, but a small 51ms difference [2ms, 101ms], $p = 0.042$). For SELECTION TIME, no pairwise differences were found.

A significant main effect was found for target DIRECTION on ERROR RATE ($F_{4,40} = 3.120$, $p = 0.025$) (Figure 3-6b). Pairwise comparisons found ERROR RATE for direction 0-forward, was 5.8% lower than direction 3-diagonal-backward, and direction 0-forward was 6.1% lower than direction 4-backward (all $p < 0.04$).

3.2.6.6 Subjective Feedback

When interviewed, the majority of participants (11) reported using their toe to tap, 2 participants reported using their heel for at least some of the targets, and 2 only participants reported using the whole foot on some targets. No participants reported significant fatigue or discomfort. 8 experienced some minor fatigue or discomfort at some point and 3 reported no discomfort at all.

3.2.7 Discussion

Although task time is significantly affected by all variables except FOOT, the effect size is small – differences in time were generally less than 25%. Distance has the greatest effect on SELECTION TIME and ROUND TRIP TIME, increasing both on the order of 100ms for a 7.5cm increase in distance. With many repetitions, these small time differences may add up, but error rate has the largest effect on usability due to additional costs from user frustration and mistaken actions. We consider error rate to be the most important factor.

The most significant factor influencing error rate is target size. Targets with angular size less than 22.5° or radial size less than 5cm had error rates in excess of 20%, and should be avoided in a real system. The best target size considering both ERROR RATE and ROUND TRIP TIME was the largest (radial 20cm, angular 45°) with an error rate of 3%.

Our results show tapping forwards is easiest and the backwards and backwards-diagonal directions somewhat more difficult. There were moderate differences in ERROR RATE (about 5%) and very small differences in ROUND TRIP TIME (about 50ms).

There was no significant main effect for feet for the right-footed participants in this study, and 95% confidence intervals indicate the possible effect size is small. Foot dominance is not an important consideration in foot interaction.

With no reports of significant fatigue or discomfort in this 60-minute rather intensive experiment, we believe that 60 minutes may be a reasonable upper bound for continuous discrete foot input.

3.3 Experiment 2: Kicking and Feedback

Building on the results of Experiment 1, we designed the second experiment to compare the different styles of pointing action, tapping and kicking, and to test performance with no cursor feedback. User ability and preference between tapping and kicking would influence which technique would be used in a real system. If both types of technique are useable, then both of them could be used for different commands, allowing a greater system command throughput. Including a no feedback condition tested the feasibility of eyes-free foot input, where indirect cursor feedback is not available or the user's visual attention is focused elsewhere. Feedback effects are important because they could constrain the domains in which the interaction technique could be applied if the user must look at a screen to use the technique effectively. Additionally, a high requirement for feedback would make it difficult to perform cognitively intensive tasks, as the user would need to split their attention between the feedback mechanism and the task at hand. To accommodate these additional factors, we reduced the number of target sizes by eliminating the lowest performing sizes from Experiment 1, which we were confident would not be useful in a real system.

3.3.1 Participants

Twelve participants were recruited (5 female), ranging in age from 20 to 30. 12 reported they were right-footed, and 7 reported they previously used a whole body input device.

3.3.2 Apparatus

We used the same apparatus as was used as Experiment 1, described in Section 3.2.2 (Apparatus), but modified the pointing action detection algorithm to robustly detect both tapping and kicking. To accomplish this, the algorithm ignored height, using speed and direction of travel only. Specifically, a pointing action was triggered when foot speed fell below 0.2 m/s, or direction of foot travel reversed along a vector from home target to task target. To avoid hysteresis, foot speed had to be greater than 0.3m/s and the foot had to move away from the home target before a pointing action was triggered. This detection method reliably detected slow or rapid taps and kicks, although it did not detect whether the performed action was a tap or a kick.

Using an under-constrained detection algorithm had multiple benefits. First, it simplified the system and reduced unnecessary system errors. Second, it allowed participants to adopt a wider range of movements and pointing strategies that could inform system design. Third, we gather more representative data of tap and kick actions that could be mined in the future to tune the design of a tap or kick specific sensing algorithm. During the experiment, participants were instructed to perform either taps or kicks and the experimenter monitored their adherence (significant differences found in SELECTION TIME suggest that the participants did adequately adhere to the experimenter instructions).

3.3.3 Tasks and Stimuli

The task and stimuli were the same as Experiment 1, but with a reduced subset of target size and distance variations to accommodate the new extra factors of POINTING ACTION and FEEDBACK.

3.3.3.1 Pointing Action

Two types of POINTING ACTIONS were tested: TAP and a midair short KICK. To complete the task using the KICK action, the participant lifted one foot off the center home target, moved their foot in the air until the foot cursor hotspot was over the task target, and reversed direction to select it. They immediately returned their foot to the home position, tapping the floor with the foot cursor inside the home target. This cycle was repeated 3 times in rapid succession for the same foot and task target. The new detection algorithm also permitted the exact same TAP pointing action as Experiment 1 with either heel or toe taps.

3.3.3.2 Feedback

The FEEDBACK condition used the same red dot foot cursor as Experiment 1, but in the NO FEEDBACK condition this cursor was hidden. NO FEEDBACK was tested with both TAP and KICK pointing actions. Targets were shown with post-selection feedback in both conditions, and the change in color when a target was activated faded out over a brief period of time, rather than disappearing immediately. Error feedback was also made clearer with a soft error sound, and sounds for target selection.

Hiding the cursor in the NO FEEDBACK condition establishes if it is possible to have foot interaction occur without a person looking at a feedback display. They would only need prior knowledge of the position of targets they wish to activate and receive feedback resulting from the system action they selected.

3.3.3.3 Target Size and Distance

For the FEEDBACK condition, target SIZE was limited to three (A, R) pairs used in Experiment 1: A2R4 = (22.5°, 20 cm), A4R4 = (45°, 20 cm), and A4R2 = (45°, 10 cm). These were used in 4 combinations with two DISTANCES (7.5 cm and 17.5 cm): A4R4 at a distance of 7.5cm, A2R4 at distance of 7.5cm, A4R2 at a distance of 7.5cm, and A4R2 at a distance of 17.5cm. These combinations correspond to tasks from Experiment 1 with reasonable error rates. The far DISTANCE was increased to exactly simulate a system with two concentric, non-overlapping rings of 10cm radial size targets (since the 5cm radial size was eliminated due to high error rate).

For the NO FEEDBACK condition, we only used A4R4 at a distance of 7.5cm, as this task had the lowest error in the previous experiment and is the easiest target to select.

3.3.4 Design and Protocol

The independent variables are FOOT (left or right), target SIZE, target DIRECTION, target DISTANCE, POINTING ACTION (TAP or KICK), and FEEDBACK (FEEDBACK or NO FEEDBACK).

Tasks were divided into 10 target configuration sets of 10 tasks. Each set covered all values of 5 target DIRECTIONS and FOOT for one combination of SIZE, DISTANCE, POINTING ACTION, and FEEDBACK. All target configuration sets were presented in random order as one BLOCK. Participants completed three BLOCKS in order to test for learning effects. After the experiment, participants were asked to demonstrate their comfortable range-of-motion for tapping and kicking and complete a post-experiment questionnaire for subjective ratings for tapping and kicking and feedback and no feedback. The experiment took 60 minutes on average.

In summary the design was:

3 BLOCKS ×

2 POINTING ACTIONS ×

2 FEET × 5 target DIRECTIONS ×

(1 SIZE and DISTANCE with NO FEEDBACK

+ 4 SIZE and DISTANCE combinations with FEEDBACK) ×

3 repetitions of serial selections

= 900 data points per participant

3.3.5 Analysis

The primary dependent variables are the same as Experiment 1: ERROR RATE, SELECTION TIME, and ROUND TRIP TIME. We introduced a secondary dependent variable to characterize pointing action characteristics: DWELL TIME is the time that the task target was activated. For the sensing system used in the experiment DWELL TIME consisted of the time that foot speed was below the 0.3 m/s threshold.

3.3.5.1 Outliers

Trials times more than 3 standard deviations from the mean for a target configuration were removed, as in Experiment 1. This removed 88 out of 10800 (0.81%) SELECTION TIME data points, 61 out of 10800 (0.56%) ROUND TRIP TIME data points and 179 out of 10800 (1.66%) DWELL TIME data points.

3.3.6 Results

All main effects use a repeated measures ANOVA, all post hoc tests use a Bonferroni adjustment. A Greenhouse-Geisser correction is applied in the ANOVA where Mauchly's Test of Sphericity is significant, and corrected degrees of freedom are reported.

3.3.6.1 Learning Effect

A significant effect was found for BLOCK on SELECTION TIME ($F_{2,20}$=7.530, p=.003), ROUND TRIP TIME ($F_{2,20}$ = 14.947, $p < .001$) and ERROR RATE ($F_{2,20}$ = 7.390, p = 0.004). Post hoc pairwise comparisons revealed a significant ($p < 0.03$) difference in all three conditions between the first and third block. Block 1 was discarded as in the first experiment to reduce learning effects.

3.3.6.2 Foot

No significant main effect was found for FOOT on ERROR RATE ($F_{1,11}$ = 1.758, p = 0.212) or for FOOT on SELECTION TIME ($F_{1,11}$ = 0.956, p = 0.349), but a significant main effect was found for FOOT on ROUND TRIP TIME ($F_{1,11}$ = 7.624, p = 0.019). Pairwise comparisons showed that the left foot was slower than the right foot, but only by 24ms [5ms, 42ms].

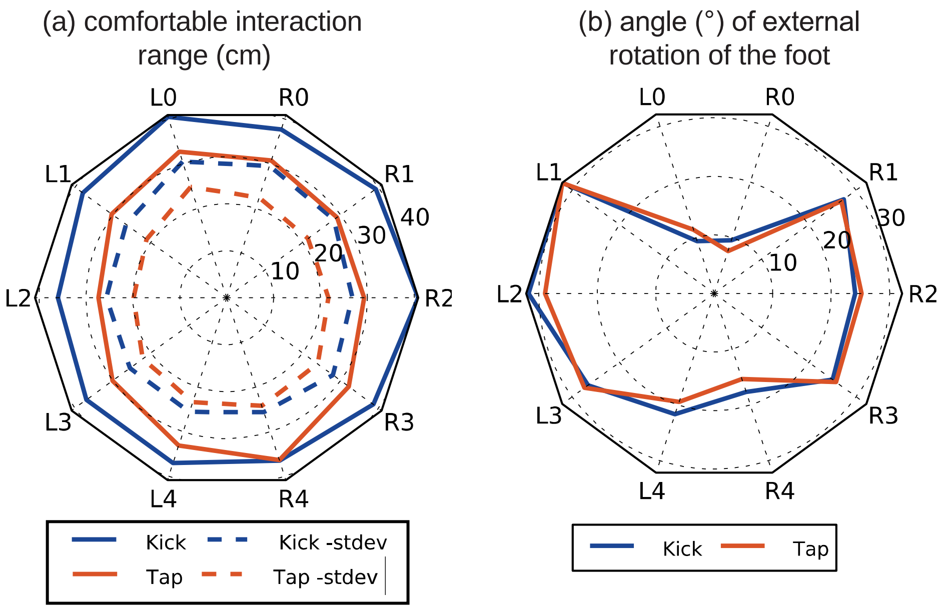
3.3.6.3 Pointing Action

Analysis of POINTING ACTION is only applicable to tasks with FEEDBACK. A significant main effect was found for POINTING ACTION on SELECTION TIME ($F_{1,11}$ = 17.642, p = 0.001). Pairwise comparisons showed TAP is 34ms [16ms, 52ms] faster, taking 247ms compared to 284ms for KICK. There was also a significant difference for POINTING ACTION on DWELL TIME ($F_{1,11}$ = 42.964,

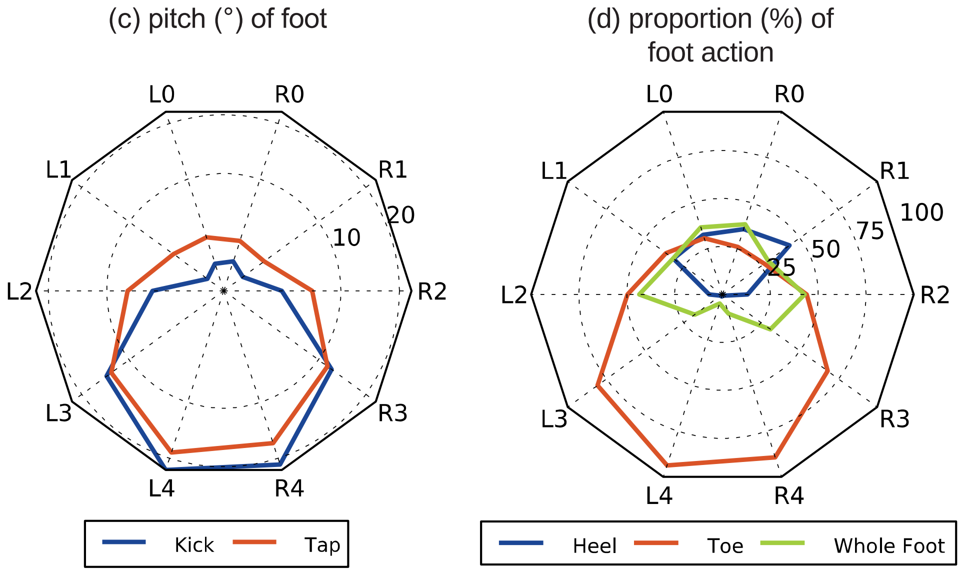
$p < 0.001$). The mean difference was 187ms [124ms, 250ms], with KICK having the lowest dwell time of 121ms vs. TAP with a DWELL TIME of 308ms. There were no significant effects of POINTING ACTION on ROUND TRIP TIME ($F_{1,11} = 0.961$, $p = 0.348$) or POINTING ACTION on ERROR RATE ($F_{1,11} = 0.152$, $p = 0.704$).

3.3.6.4 Feedback

We compare FEEDBACK and NO FEEDBACK using target A4R4 at distance 7.5cm. A significant main effect was found for FEEDBACK on ERROR RATE ($F_{2,11} = 28.859$, $p<0.001$). NO FEEDBACK had a mean error rate of 27.5%, while FEEDBACK had a mean error rate of 4.4% (mean difference 23.5%, [32.6%, 13.7%]). No significant effect was found for FEEDBACK on SELECTION TIME ($F_{2,11} = 0.274$, $p = 0.611$), or on ROUND TRIP TIME ($F_{2,11} = 3.604$, $p = 0.084$) and no significant interaction effects were found involving FEEDBACK.


(a) comfortable interaction range (cm)

(b) angle (°) of external rotation of the foot


(c) pitch (°) of foot

(d) proportion (%) of foot action

Figure 3-7. Comfortable interaction range and foot motion characteristics (a) Comfortable range-of-motion in cm for tapping and kicking as demonstrated by participants; (b) angle of external rotation of the foot in degrees; (c) pitch of foot in degrees; (d) proportion of heel, toe, or whole foot actions based on Vicon log analysis

3.3.6.5 Comfortable Interaction Range

At the end of the experiment, we asked participants to demonstrate the maximum range they would be comfortable interacting at with both of the techniques. Using the Vicon tracker, we processed these demonstrations into range of motion by distance (Figure 3-7a). We include the mean value and the mean value less one standard deviation as a more conservative estimate. The comfortable range of motion is 30cm for TAP on average, and for KICK it is roughly 40 cm on average, falling to 35 cm in the backwards direction. Reducing the area by one standard deviation yields a conservative estimate of 20cm for TAP and 30 cm for KICK.

3.3.6.6 Foot Rotation and Pitch

We calculated the external rotation and pitch of the foot using the recorded Vicon data. *External rotation* is the outwards angle of foot rotation around the heel axis relative to the foot pointing forward (if rotated inward, it would be negative external rotation). *Foot pitch* is similar to plantar flexion but the angle is measured relative to the floor, as the angle of the heel-to-toe vector above the floor plane. Lowering the toe relative to the heel increases foot pitch (raising the toe relative to the heal leads to negative pitch). Participants externally rotated the foot more on side and diagonal targets (Figure 3-7b), and lifted the heel more on backwards targets (Figure 3-7c).

3.3.6.7 Heel or Toe

Since our target selection algorithm does not depend on whether the front or back of the foot is near the floor, we can examine the data to see how participants naturally tap. For this analysis, we classify tap type using the foot pitch at selection time. If foot pitch > 5°, we consider it a heel tap; if foot pitch < -5° it is a toe tap; otherwise a whole foot tap. Using this metric, participants tapped with their heel in 17% of the trials, their whole foot in 29%, and their toe in 54%. The distribution of tap type by foot is near-symmetric, but there are proportionally more toe taps to backwards targets (Figure 3-7d)

3.3.6.8 Post-Experiment Questionnaire

At the end of the experiment, we gave participants a questionnaire, included in Appendix A. We asked 12 questions about the ease of use of the tap and kick for the FEEDBACK and NO FEEDBACK conditions.

	It is easy to perform mentally	It is easy to perform physically	It is easy to learn
Tap with Feedback	1.5 (0.7)	1.6 (0.7)	1.3 (0.5)
Kick with feedback	1.7 (0.7)	2.2 (1.0)	1.7 (0.9)
Tap without feedback	2.2 (1.2)	2.4 (1.2)	2.4 (1.1)
Kick without feedback	2.9 (1.4)	3 (1.2)	3 (1.0)

Table 3-2: Results of participant questionnaire
The Likert scale used was: (1) Strongly Agree (2) Agree (3) Neither
(4) Disagree (5) Strongly Disagree
Averages are reported, with standard deviation in parentheses.

Two additional questions were asked to directly compare tap and kick on a 5 point Likert scale (1 = strongly prefer kick, 5 = strongly prefer tap), controlling for level of feedback.

Between kick with feedback and tap with feedback, which do you prefer? = 3.8 (stdev = 1.4)

Between kick without feedback and tap without feedback, which do you prefer? = 4 (stdev = 1.0)

These results suggest a slight preference for TAP over KICK, and looking at the individual questionnaires 8 participants preferred or strongly preferred TAP over KICK as an interaction technique. However, 9 participants agreed or strongly agreed that KICK was easy to perform physically (average score 2.2), 11 participants agreed or strongly agreed that KICK was easy to learn (average score 1.7), and all participants agreed or strongly agreed that KICK was easy to perform mentally (average score 1.7). Several participants reported that they liked the tactile feedback provided with TAP, felt it improved their performance, or felt like it was less effort. P11 reported that "Tapping was slightly less tiring, probably important if I were doing this for a long while." P7 reported that for TAP that it "lets you ground your foot to regain balance/stability [and] provides a reference for tapping foot in the same place next time." And P5 remarked that "having that feeling of tapping onto the ground just makes me feel better."

Some participants reported positive aspects of KICK. P8 felt that “kicking is more intuitive, fun; tapping is an additional chore.” P1 reported “kick is easier. [It] Costs less energy, feels comfortable.” P2, although preferring tap, felt that “tap has a more definite feeling, but kick is faster.”

Similar to Experiment 1, 5 participants reported minor discomfort, 3 of them attributing the discomfort to either KICK or KICK with NO FEEDBACK.

3.3.7 Discussion

We did not find a large quantitative difference between tapping and kicking pointing actions in time or error rate measures. Participants showed an overall preference for tapping, but kick was not rated poorly on an absolute scale. This suggests that tapping and kicking are both suitable for use in the proposed interaction technique, but that more frequent actions should be assigned to tapping.

The lower SELECTION TIME for tapping may be due to how the foot rapidly decelerates when it contacts the floor, while kicking requires the foot to decelerate using only leg muscles. The faster tap selection time occurs in spite of tap having a larger DWELL TIME. This difference in tapping and kicking motion characteristics might be exploitable in producing a detection algorithm which can discriminate between tapping and kicking (although we ended up using a different method for our system).

With an error rate above 20%, the NO FEEDBACK condition as tested may be infeasible for use in a real system. However, the results left open a possibility of refining the feedback method to improve this error rate to acceptable levels. A simple approach to this is increasing the target radial and/or angular size, though this decreases the command throughput of the system. While running the study, we observed that errors in the NO FEEDBACK condition seemed to be in part due to situations where the participant loses track of the center of the home target (even while their foot landed within the target), and hence the relative position of the target. This can mitigated by shifting the home target to coincide with where the user’s foot lands within the home target or only considering motion relative to position of the user’s foot at the start of the motion. Another option is to provide a feedback display that is only opened or glanced at when errors occur, but can be ignored most of the time. We describe the methods we use to address this issue in Tap-Kick-Click in Section 4.1.1.2.

3.4 Discussion and Design Guidelines

3.4.1 Comparison to Previous Work

A direct comparison with kicking contextualizes different interaction options and previous related directional kicking work from Han et al. [14] and Alexander et al. [1].

Han et al. [14] recommend an angular target size of 24 degrees for directional kicking with a 12% error rate and Augsten et al. [3] recommend rectangular 5.3 × 5.8 cm targets for tapping, but both tested the dominant foot only. The goal of our design evaluations was to confirm these results for tapping and kicking with dominant and non-dominant feet, but within the context of our interaction technique: using radially distributed virtual targets and under pure indirect control (with and without a foot cursor). With indirect feedback, we found that tapping on angular target sizes of 22° or greater had error rates less than 10%, similar to Han et al.'s recommendation for kicking [14].

We also confirm the importance of direction for interaction, replicating the finding of Alexander et al. [1] that backwards kicks were more difficult.

Garcia and Vu [11], although investigating a foot mouse which is very different from the interaction technique we investigated, found a learning effect for foot input over multiple sessions and which was greater than that for hand input. This is in line with the learning effect found in this study, and suggests that performance might significantly improve if participants were able to practice over a longer period of time.

We found little support for a performance difference between feet, with only 24ms in ROUND TRIP TIME in Experiment 2. This means the user preference to alternate feet found by Meyers et al. [27] is supported without increased time or errors.

3.4.2 Design Guidelines

Based on experiment results, we proposed ten design guidelines relevant to our proposed interaction technique, and to the general space of indirect foot pointing:

G1. Tapping and kicking are both feasible, but users have a slight preference for tapping: use tapping for more frequent actions.

G2. People use both feet equally well; any effect of foot dominance is small.

G3. When tapping, people prefer toe taps. Use toe taps for most common actions, then whole foot taps, then heel taps.

G4. All of the investigated target directions are feasible. Forward movement is less error prone to use, and backwards and backwards-diagonal interaction are hardest to use.

G5. For indirect cursor feedback, target angular size should be at least 22.5°; two target levels is feasible with radial size 10cm.

G6. Without cursor feedback, a target angular size of 45° and 20 cm is insufficient. It may be possible to improve performance with a larger target size, or different feedback techniques.

G7. Increasing distance of targets within reasonable limits increases interaction time, but does not increase error.

G8. A conservative estimate for an appropriate interaction radius is 20 cm for tap interaction, and 30 cm at the front and 25 cm in radius at the back for kick interaction.

G9. 60 minutes of continual foot interaction, with occasional breaks, is feasible for users to do with only minor discomfort.

G10. Sensing techniques must be robust to changes in foot pitch and external rotation of the feet with sideways motion.

These guidelines inform design of the system presented in Chapter 4.

Chapter 4

Tap-Kick-Click

After conducting the initial experiments described in Chapter 3, we proceeded to refine the interaction technique described in 3.1. The design guidelines from 3.4.2 enabled us to refine the specifics of the system, such as which foot actions and directions to use and the size of the interaction area. With these details finalized, we set out to implement a full foot input system for controlling applications. We were able to create a new sensing system which relied on a low-cost Kinect camera and Arduino-based sensing platform instead of the expensive Vicon motion capture system, and provided greater accuracy in foot action discrimination. We also developed a user interface designed to provide an appropriate level of feedback and guide the user through controlling a real world application. We also designed a command mapping for the system to use in turning foot input into control of a web browser. Together, these constitute a foot input system which can be used to control real-world applications. We dubbed the system “Tap-Kick-Click”, referring to the tapping and kicking foot actions used in the system and the “Click Mode” used for selecting GUI targets (described in 4.1.2.3). Finally, we performed an evaluation of the system, introducing new users to the interaction technique and asking them to perform tasks in a web browser. We found that users were able to learn the basics of the system in a 60-minute study session, and gained useful insight into which components of the system were difficult to use or understand.

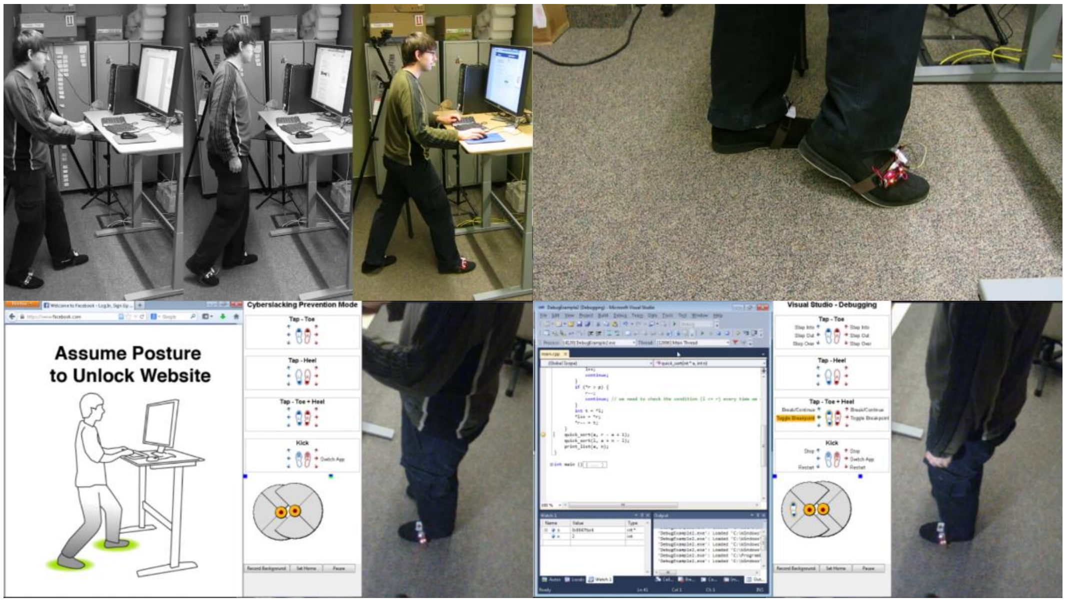


Figure 4-1: Video Figure for Chapter 4
This chapter is accompanied by a video figure, which accompanied our submission to the UIST conference to demonstrate the features and use of Tap-Kick-Click. It is available at:
http://williamsaunders.net/thesis2015chapter4 or
http://youtu.be/E1BswhGCCmY

4.1 Interaction

To control a desktop application using foot input, the Tap-Kick-Click system needed to implement not only an interaction technique, but also a means for sending commands to the application, a mapping or correspondence between foot actions and commands in the application, and a method for indicating to the user which foot actions correspond to which commands.

4.1.1 Interaction Technique

The core interaction technique used in Tap-Kick-Click is fundamentally the same as the one described in Section 3.1, with some of the details specified based on the results of the quantitative studies described in Chapter 3.

4.1.1.1 Foot Actions

We decided to use all four of the following foot actions (Figure 4-2): toe (front of the foot, with either the toes or ball touching the ground), heel, and whole foot (where both the front and back of the foot are touching the ground) and kick (moving the foot over a target and back to the home position). A tap selects a target when and where the foot contacts the ground while the foot cursor is detected to be over the target and a kick selects a target when and where the foot reverses direction over the target (as though it was kicking a virtual ball). Between each tap and kick, the foot typically returns to a central home position to maintain balance and encourage physical activity, although multiple taps can take place on the same or different targets.

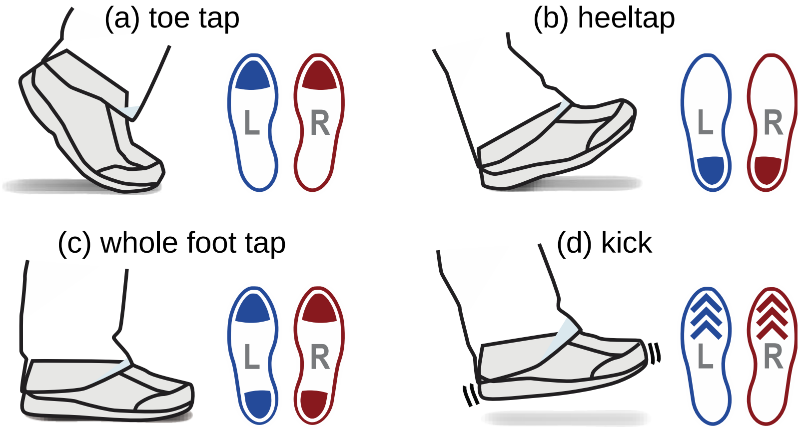


Figure 4-2. Tap and kick types with foot-action interface icons.

This combination of foot action and target selection to create a discretized input language. Application events can be triggered from a single target selection or an ordered sequence of selections. Discrete input can be used for navigation: we scroll with a page up and page down metaphor using forward and backward taps. For long distance scrolls, tapping actions can be "auto-repeated" when the foot is held on a target for a period of time (after 150ms, the action repeats every 400ms). This technique was also used in Meyers at al. [27] for scrolling through lists of items. Any tap-activated action can be configured to auto-repeat, if appropriate. In Section 4.1.2.3 we explain how single selections are mapped to application commands and how sequences are used to select GUI targets in "click mode."

4.1.1.2 Selection Hotspot and Virtual Target Layouts

Based on the results of the quantitative studies, we continued to define a single, consistent hotspot location on the foot. Instead of the middle of the foot, Tap-Kick-Click uses the position of an LED mounted on the Arduino sensing platform, which usually was positioned in the center-front of the foot. The change was made because the sensing mechanism returns only the 3D position of a single point without orientation, and the position of the LED was limited by the geometry of the foot.

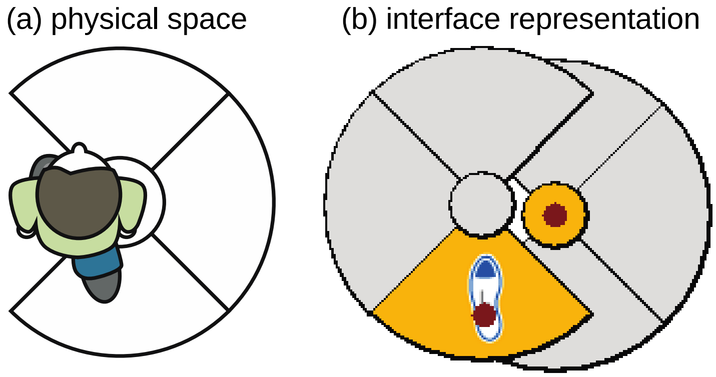


Figure 4-3. Low-density virtual targets
Low-density virtual targets: (a) physical space; (b) foot cursor feedback showing left toe tap back action.

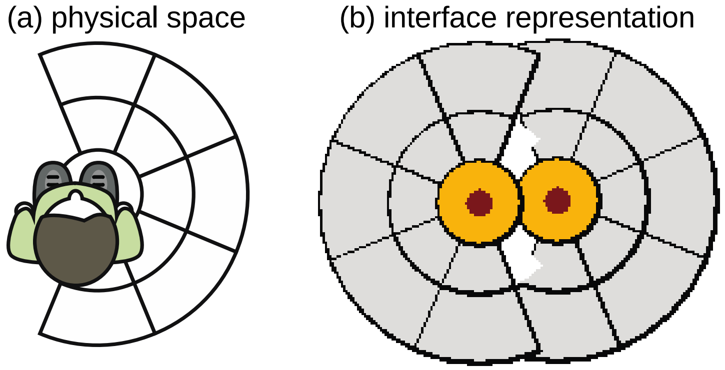


Figure 4-4. High density virtual targets
High density virtual targets: (a) physical space; (b) foot cursor feedback showing both feet at home

Despite the poor performance of the eyes-free input system tested in the initial quantitative studies (G6), we decided it was important to make the primary method of inputting application commands eyes-free. Eyes-free input means that the user can use the system without switching back and forth between looking at an application and looking at a feedback display, and can perform input actions with less cognitive load. To improve the performance of eyes-free input, we made several changes relative to the setup in the initial experiments. First, we use an always visible sidebar to present a small feedback display in the lower right corner of the screen. While the user does not need to be constantly looking at it, they can glance at it if they need to troubleshoot a command that was performed or recognized improperly. Second, we made the targets significantly larger than in the no feedback condition of Experiment 2 (3.3), with the low-density layout shown in Figure 4-3. The targets are positioned at a distance of 10cm from the home position, have an angular size of 90º and a radial size of 40cm. Finally, we implement an adjustment to the home position to deal with situations where the user's foot does not return to the exact center of the home position. The center of the feet home locations is adjusted once each second to match the actual foot positions when the foot is stationary and within a 5cm radius from the previous home position. Together, these tweaks made it possible to use indirect foot input with reasonable reliability (at some cost in command throughput).

Additionally, our system provides a high-density target layout (Figure 4-4) with size based on the recommendations for targets with indirect cursor feedback (G5) and attempting to stay close to the conservative estimate for an appropriate interaction radius (G8). The high-density targets have an angular size of 45º and are positioned in a contiguous outward arc from front to back of each foot in two bands, one from 10 to 20 cm and the other form 20 to 30 cm (Figure 4-4a). This high-density layout is used for the special-purpose foot menu where the user can fully rely on indirect feedback to select among many simultaneous actions.

4.1.1.3 In Place Jumps and Standing Postures

Figure 4-5 Mildly uncomfortable foot position used to discourage cyberslacking.

We supplement this core language of taps and kicks with in-place two foot jumps and static standing postures sensed by foot position. Meyers et al. [27] report that people find two foot jumps enjoyable, but using them for core application control seemed too physically demanding. We use two foot jumps to activate a help system since we thought that jumping action would be easy to remember and the high-energy movement could encourage learning (to avoid this exertion by consulting help too frequently).

We use static postures as a kinesthetic mode to unlock websites flagged as sources of cyberslacking (websites where users might waste time during work hours e.g. Facebook, YouTube, etc.). We use a mildly uncomfortable posture resembling a lunge, where one foot is back and the other forward such that the knee is bent and thigh muscles are under tension (Figure 4-5).

4.1.2 Application Control

Rather than design new applications specifically for foot input as done in previous work [27], we designed our system to control real applications. The Tap-Kick-Click interaction vocabulary can accomplish useful application tasks, but our philosophy is not to attempt 100% control. There are certain tasks that would be tedious with pure foot input, such as drawing or long periods of text entry and target selection. Since we combine foot input with a standing desk, standard mouse and keyboard input are always available to handle these cases. Our aim is provide enough control that feet can be used as the sole input method long enough to take breaks away from the keyboard. By providing a reasonable level of control, we also create opportunities for increased physical activity when feet augment mouse and keyboard input by triggering foreground and background application commands.

4.1.2.1 Mapping Foot Actions to Commands

A foot action on a virtual target in the eyes-free low-density layout is the primary way to trigger application commands. Each time a tap or click is sensed, the corresponding command in the current set of mappings is sent to the application by injecting a keyboard shortcut key sequence. With 4 foot actions (toe tap, heel tap, whole foot tap, or kick) and 6 virtual target locations across both feet, 24 commands can be accessed in one set (see examples of command sets in Figure 4-11). We avoided use of two particularly difficult commands, heel tap backwards with left and right foot, bringing the total commands down to 22. The default set of mappings is synchronized with the current foreground application and the state of the application, using information in the window title text. We define the command sets using a set of application specific configuration files, but this could be partially automated using accessibility APIs to scrape application menus. Sending commands to applications can be implemented using a variety of methods, although we found that simulating keystrokes provided most of the functionality we required. We implemented keystroke macros using a dll wrapper to the Autohotkey macro program [51].

To handle more than 22 commands, some kick actions are used to change command sets, analogous to a multi-level menu. For example, an outward right-foot kick always changes to a command set for application switching and background application control. In this set, toe taps can be used to access frequently used applications, and a forward heel tap switches to the previous application. In the same set, preconfigured background applications can also be controlled, for example a whole foot tap to the right sends ‘play’ to a music player and a right whole foot tap up skips to the next track. Changing command sets explicitly is not always required since application context will often suffice. For example, in our programming IDE application, the debugging control command set (Figure 4-11d) is automatically activated when the IDE enters an interactive debug session.

Where possible, we map frequent commands to forward targets and favour toe taps over other forms of tapping and kicks, since our initial study found that these were preferred (G1, G3, G4). However, highly correlated mappings overrule this guideline. For example, we initially assigned the scroll-down command to a forward tap since this is a frequent action, but iterative user testing revealed a strong dislike for this opposite mapping. Alexander et al. [1] also note the importance of correlated mappings. We also distribute common commands across both feet to balance their use as much as possible. For frequent commands, such as scroll down, we map the same command to both feet so the user can self-balance. Our design studies found that backward heeltaps are difficult, so this combination should be avoided when possible, or assigned irreversible commands (e.g. “delete all”) to avoid accidental invocation. The mapping we present here is preliminary – it may be possible to further improve the mapping based on observations of command frequency over a period of real-world use of the system.

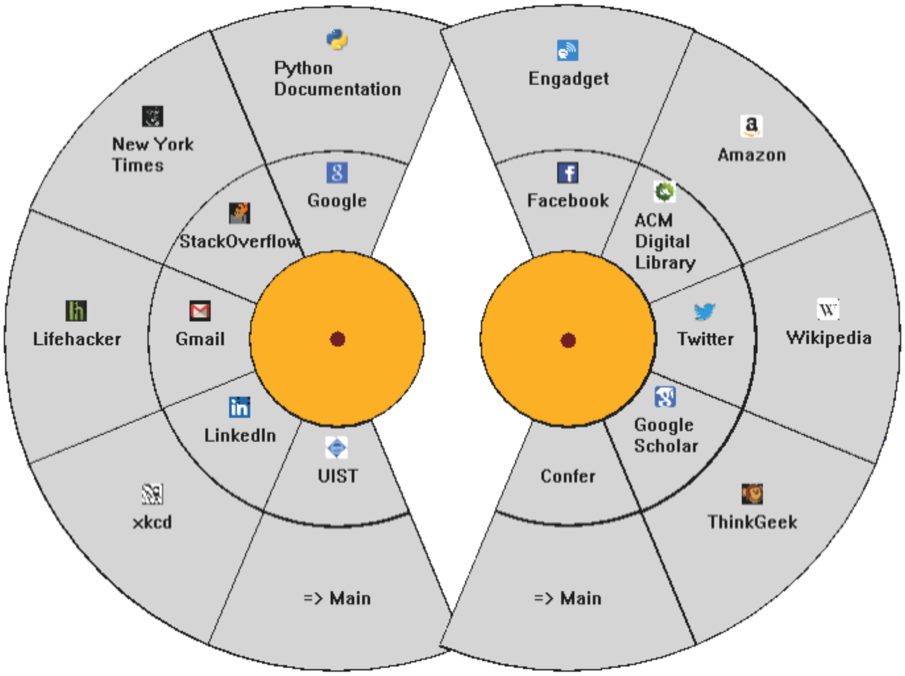


Figure 4-6. Foot menu using high-density layout

4.1.2.2 Foot Menu

The high-density layout is used for a special-purpose foot menu (Figure 4-6). When the menu is invoked, the virtual targets and foot cursor location are shown in the centre of the screen. This central indirect feedback is what enables the higher target density. For example, a left kick in our web browser command set opens a foot menu to load a new webpage from a list of favourite bookmarks. Each target is labelled with a menu action. With 4 different foot actions and 20 virtual targets, 80 menu actions are possible, although we limited the system to only 20 commands during the evaluation for ease of understanding the menu display.

4.1.2.3 “Clicking” on GUI Targets

There are many cases in GUI applications where arbitrary targets must be selected and there is no direct keyboard mapping, such as when clicking on web page hyperlinks. Given the poor performance of foot-controlled mice and our focus on a discrete input vocabulary, we provide a special “click mode” that uses a sequence of directional taps to select arbitrary GUI targets like hyperlinks. In our current system, click mode is activated with a forward kick when web browsing. Once activated, we use a modified version of the “Hit-a-Hint” Firefox add-on [52] to label each link in the visible part of the page with unique sequences of six actions: forward, outward, and backward taps using either the left or right foot. We display sequences beside each corresponding target as strips of 16px arrow icons (Figure 4-7). With dense targets, the arrows occlude other content, but we intend click mode to be activated after a target is mentally selected, enabling the correct sequence to be quickly located.

This scheme scales well, as sequences up to length 3 can index 258 targets $(6 + 6^2 + 6^3)$. Since sequence lengths vary, we use a forward kick to accept the entered sequence and "click" on the target. This technique could easily be extended to select targets in general GUI applications using accessibility APIs to determine the location of relevant targets, or detecting targets using a system like Prefab [9]. Clearly click mode is not as fast as using a mouse, but it is easy to learn and requires fun combinations of steps that fulfill our goal of physical activity.

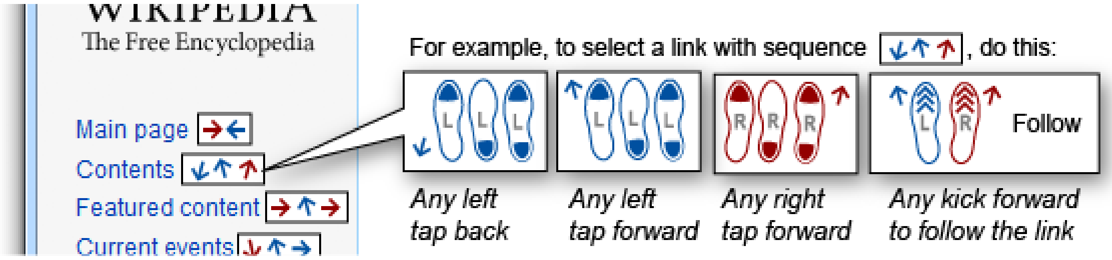


Figure 4-7. *Click mode* selection sequences

4.1.2.4 Text Entry

Augsten et al. [3] report that people can type with one foot on a 52.0 × 23.2cm interactive floor keyboard at approximately 10 words-per-minute with a 3% target error rate. We found a similar level of performance with our high-density target layout, so in theory we could include text entry. In practice, typing at this speed and accuracy is tedious: text entry is challenging with purely foot input. Our command mappings and demonstration applications show that many productive tasks can be accomplished without any text input. If some text entry is needed, the standing desk context enables the user to easily reach out for the keyboard. However, we also explored speech recognition augmented with foot input. We created a speech recognition application command set so that foot actions could flag recognition mistakes and select from an n-best list of alternative text recognitions.

4.1.2.5 Anti-Cyberslacking Postures

To encourage spending less time on distracting websites, the system detects when a website identified as distracting is loaded, and blocks the content area with a message to perform a lunge-like anti-cyberslacking pose to access the site. Once they assume the pose (Figure 4-5), the blocking message is removed. The unlock pose consists of standing with one foot forward and one foot back (either foot can be forward, and the posture can be switched at any time).

4.1.3 Learning and Reinforcement Feedback

Feedback is important in our system for two reasons. First, the user needs to be able to see and understand what the state of the system is at any given point in time. This is in order to be able to spot and troubleshoot errors (e.g. an object in the background is disrupting foot tracking). Second, the user needs to be able to learn the mapping between foot actions and application commands.

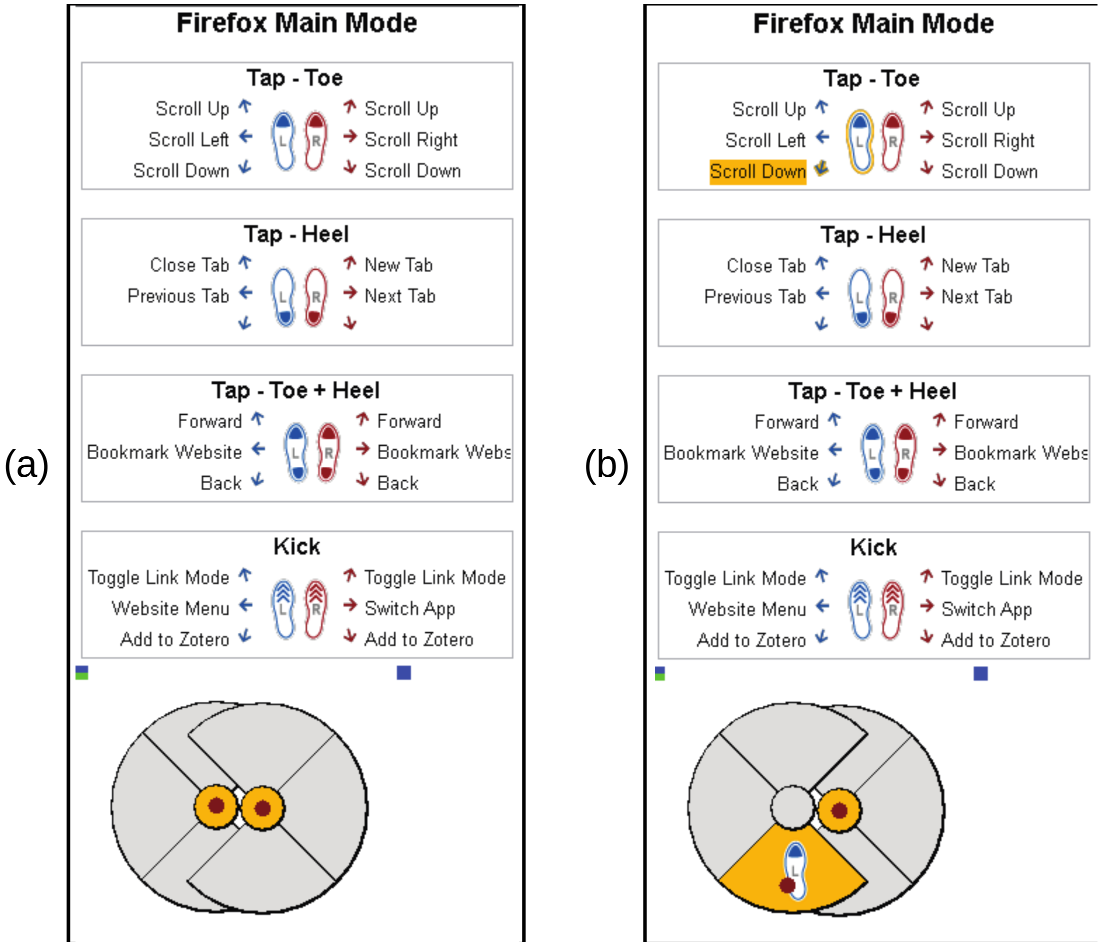
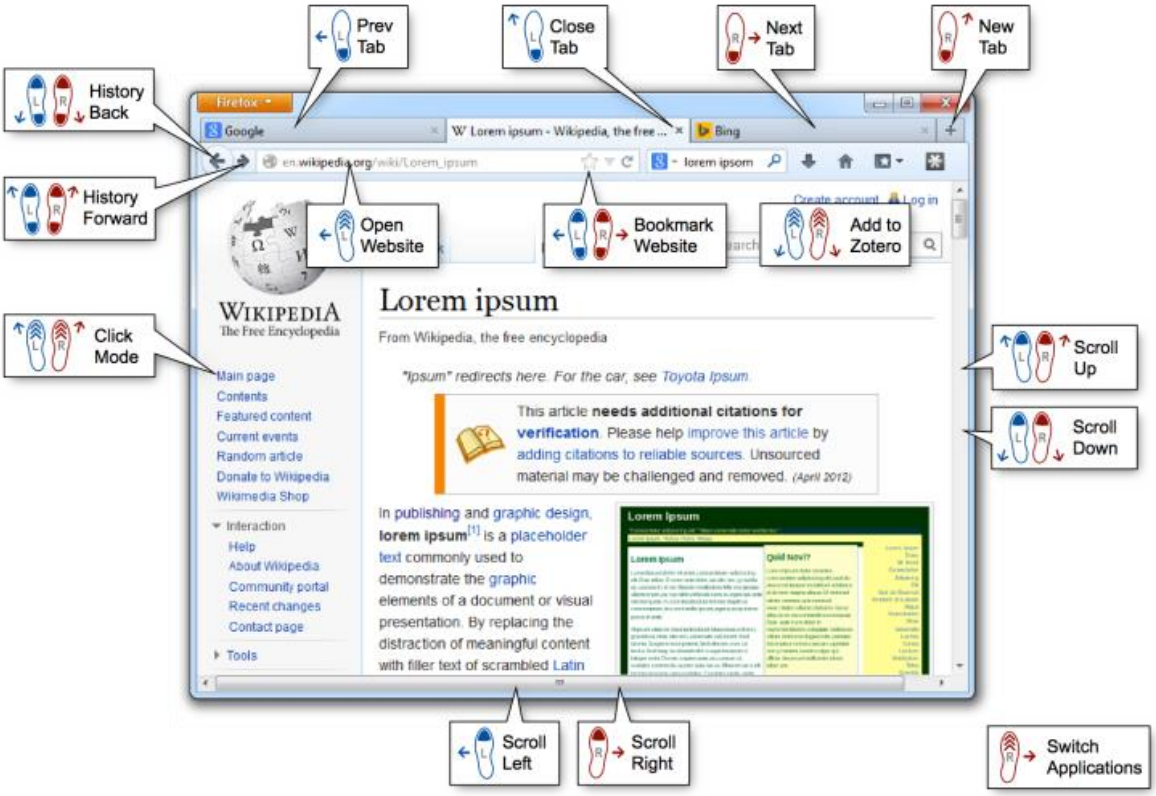


Figure 4-8. Tap-Kick-Click command and feedback sidebar
Always-visible sidebar with foot cursor and targets below and command cue card above: (a) showing neutral state; (b) showing feedback after triggering a command.

To address both of these issues, we use an always-visible sidebar displaying the foot cursor with virtual targets and a cue card showing the active command mappings for foot actions (Figure 4-8). Commands are grouped according to foot, foot action, and direction for consistent interpretation. The most recently sensed foot action is displayed using a foot-action icon (Figure 4-2) in the corresponding target location and the cue card highlights the most recent command issued. The use of a sidebar allows this information to be presented without altering the original application.



Lorem ipsum

From Wikipedia, the free encyclopedia

"Ipsum" redirects here. For the car, see Toyota Ipsum.

This article **needs additional citations for verification**. Please help improve this article by adding citations to reliable sources. Unsourced material may be challenged and removed. (April 2012)

In publishing and graphic design, **lorem ipsum**[1] is a placeholder text commonly used to demonstrate the graphic elements of a document or visual presentation. By replacing the distraction of meaningful content with filler text of scrambled Latin

Figure 4-9. Help screen
Help Screen showing foot action to command mappings in the context of the application interface.

To assist new users in learning the system, we provide help screens that convey command mappings in the context of the applications (Figure 4-9). Help screens are activated with an in place two-foot jump. Foot actions are shown in a callout pointing to the equivalent application command location when possible. Currently these help screens are manually generated, but we could explore ways to generate them automatically using accessibility APIs or a technique like PreFab [9].

4.2 Sensing

We used a sensor fusion approach to realize our interaction techniques, combining the strengths of a depth sensing camera and insole pressure sensors. We use a Microsoft Kinect for Windows for tracking the absolute position of both feet using an IR led and a custom algorithm. The pressure sensors enable the system to accurately differentiate between tapping with different parts of the foot, tapping and kicking, and to precisely sense repeated tapping actions. We implemented all code (both sensing, feedback and application control) on a PC with an i7-3770 3.4GHz CPU running Windows 7, and used the Python programming language with the wxPython GUI toolkit and numpy library.

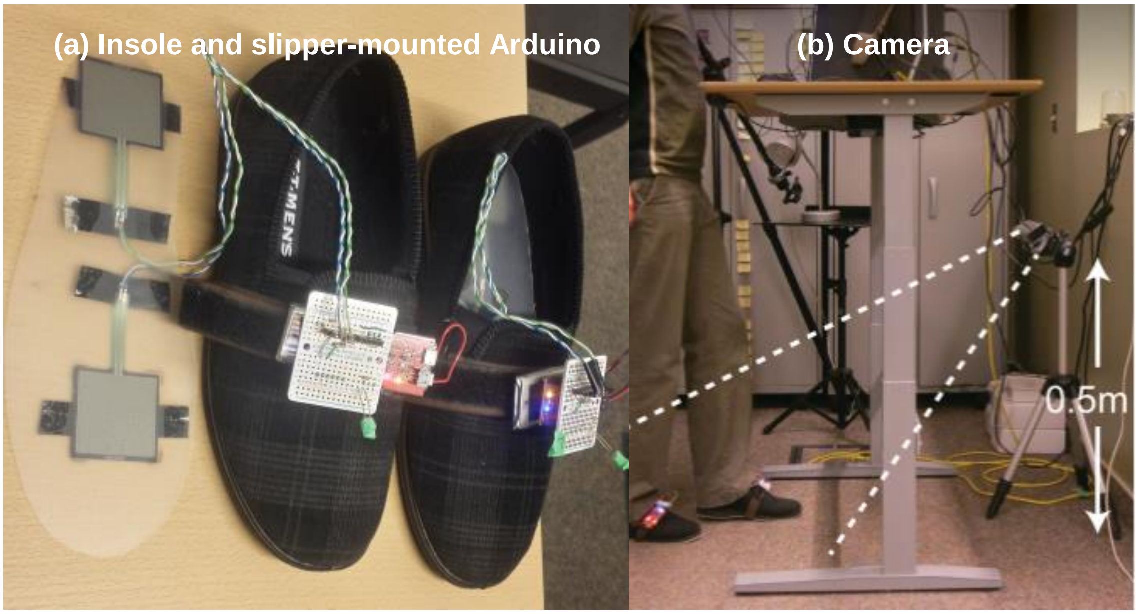


Figure 4-10. Sensing hardware
(a) insole with two pressure sensors connected to shoe mounted Arduino board; (b) under desk Kinect depth camera for foot position tracking

4.2.1 Kinect Sensing

We wanted camera placement to be unobtrusive and practical for smaller office spaces. We mounted the camera on a tripod behind the standing desk, such that when the desk is raised, the camera has a clear view of the user's feet and lower leg (Figure 4-10a). Standard skeletal tracking algorithms assume an unobstructed view of the whole body or upper body which we could not obtain in this configuration, so we mounted an 850nm IR LED on a central location on the slipper and used it to reliably track the position of the foot in real space. Tape was placed over the IR LED to diffuse the illumination so it was more consistent as the foot underwent external rotation or change in pitch (G10).

The LED tracking algorithm work as follows:

1. The current 320 x 240 px depth frame (downsampled to 160 x 120) and 640 x 480 IR frame are retrieved from the Kinect (downsampled to 320 x 240 px).
2. A 3x3 25-percentile filter is used to locate the 2 brightest locations in the IR image (Once the first location is found, a circle of radius 15 px is removed from consideration for finding the second location, to avoid double counting the same region).
3. A filter is used to select a region of the depth image around each bright spot in the IR image. The 25th percentile of depth in this region is taken as the depth value corresponding to the bright spot. The filter is the bottom half of a ring of inner radius 5 px, outer radius 15 px. This filter shape was chosen to avoid the hole in the depth image caused by the IR led illumination, and to select depth points corresponding to the slipper instead of the background.
4. The depth from the Kinect image is used to find the physical location in 3d space of the brightest points found by the algorithm. The coordinate system is calibrated by finding the floor in the known background image. The height is ignored, and the 2d position over the floor is used as the position of the foot cursor.

We found that the Kinect camera was reliably able to distinguish the LED from the background pattern of illumination it uses to calculate image depth, and that the presence of the LED in the image created only a small hole in the depth image. Tracking a fixed point on the foot made the algorithm robust to changes in foot orientation, which is important as our design studies found that users tended to rotate their foot outward when moving to the sides (G10).

4.2.2 Insole Sensing

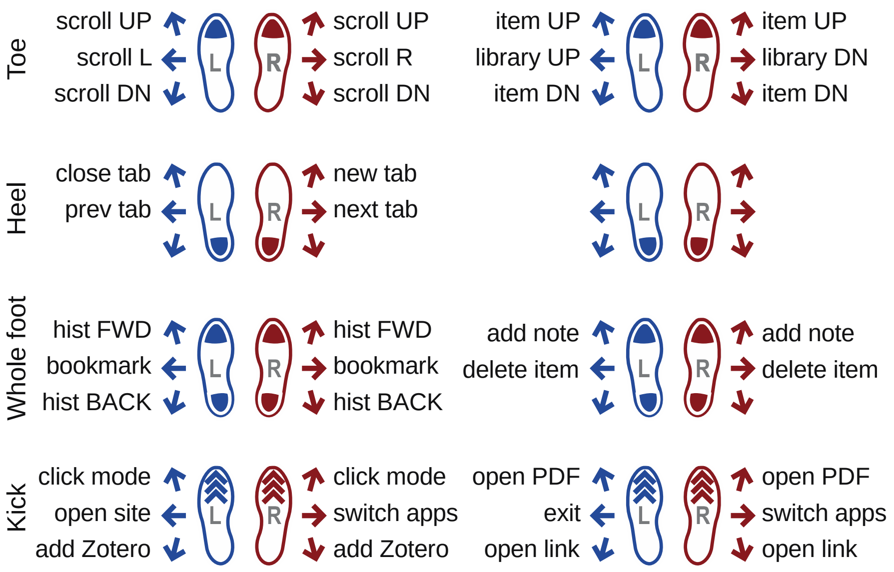
The shoe-based sensing system comprises a battery-powered Arduino to read sensors and report them to the computer via Bluetooth. The Arduino is connected to force sensitive resistors in an insole placed into the bottom of a slipper (Figure 4-10b). Pressure sensors are placed beneath the ball of the foot to sense toe tapping and beneath the heel to sense heel tapping. We found activation thresholds that were suitable for most users, but manually checked the thresholds for each new user (this process could easily be automated, but served as a useful way to spot glitches and introduce the user to the set of foot actions). Together these allow for toe tap, heel tap, whole foot tap and kick gestures to be sensed.

4.2.3 Sensor Fusion and Action Sensing

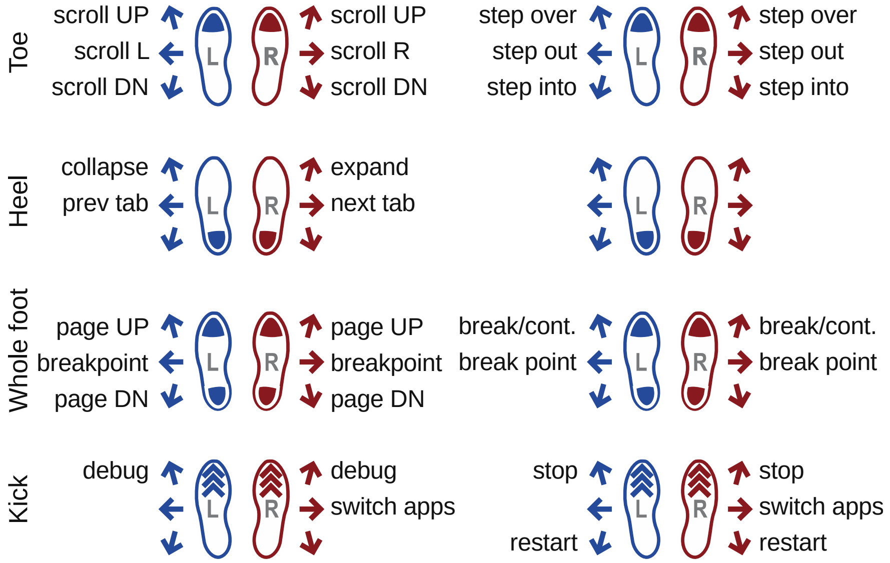
Arduino and Kinect data streams were synchronized using the time that the data was received over Bluetooth (Arduino) and the time that the depth frame was read into the programs memory (Kinect). The depth frame was loaded prior to the IR frame, as the larger IR frame took longer to process. To reliably sense combinations of foot action (from the pressure sensors) and direction (from the position data), we required an action-direction combination be sensed for a period of time (usually 100ms) before a command would be invoked. We also enforced constraints on sensing input actions to reduce false activations. For example, we assume that kick commands are only sent one at a time, and that sliding commands (where the foot is pressed on the ground while it is moved between targets) are not used.

4.3 Example Applications

To make the utility of Tap-Kick-Click concrete, we discuss three usage scenarios: general web browsing, web-based academic research, and Integrated Development Environment (IDE) debugging.


(a) Web Browsing

(b) Zotero


(c) IDE

(d) IDE Debugger

Figure 4-11. Example application mappings
Example application mappings for web browsing, Zotero, IDE control and IDE debugger control. Representation is similar to actual sidebar cue cards.

4.3.1 General Web browsing

Tap-Kick-Click can be used for the majority of tasks performed in traditional web browsing (Figure 4-11a). Although we do not provide a means for entering search queries, the system can use the foot menu to navigate through a set of frequently visited websites, or can be started after entering a query via another input method. Toe tapping is mapped to commands for scrolling within the website, link selection is accomplished through click mode, and forward and backward history navigation is mapped to whole foot commands. This provides a complete method for navigating between web pages. Heel commands are used to open/close tabs and switch between tabs. Additional commands allow for interaction with browser features, such as bookmarking a web page with whole foot outward taps. Special command mappings can be used when a specific web page is loaded, allowing for smoother control. For example we implemented (but did not use in the evaluation) a command set to make it easier to select links from a Google search result, where a kick forward with the left foot highlights the first result and allows other results to be selected by toe taps, without labelling every link on the page as in click mode. This method can be easily extended to web applications such as email or task management.

4.3.2 Web-based Academic Research

In addition to smoothly controlling one application, the system allows for the control of workflows involving multiple applications. We demonstrate this through a set of modes for controlling a web based academic research workflow, including browsing web pages, storing them in a reference manager, navigating a reference manager, and browsing PDF documents.

To switch between applications, we create a special mode for application switching and background application control, which is always accessible with a kick to the right. This mode enables access to the six most frequently used applications via a toe tap (which could be easily extended to more applications by use of a foot menu). It also provides mappings for switching to the previous application, closing the current window, and opening the task manager. Whole foot commands can be used be used for background control. As an example, we implemented a means to control a music player in the background, without needing to switch to the program itself.

The web-browsing mode includes a mapping for adding pages to the Zotero reference manager with a kick back. Command mappings for Zotero (Figure 4-11b) include toe tap commands for navigating between libraries and selecting individual items. Whole foot commands can be used to preform actions on individual items, such as adding a note or deleting the item. Kicks are used for commands that open the referenced item in a PDF viewer or a web browser. The PDF application mode, similar to the web browser, uses toe taps for scrolling and heeltaps for tab switching. Additional commands, such as zooming in or out, are kicks or whole foot taps.

4.3.3 Integrated Development Environment (IDE)

Finally, Tap-Kick-Click can be used in specialized workflows even in text-heavy applications, such as an IDE. The system includes a command mapping for navigating the IDE (Figure 4-11c), using toe taps to scroll and heeltaps to switch tabs and expand and collapse sections of code. A whole foot tap to the left or right is used to set a breakpoint, and a kick forwards can be used to begin debugging. When a debugging session starts a specialized debugging command set becomes active (Figure 4-11d). This uses toe taps to perform the classic debugging control commands to *step into*, *step out of*, and *step over* specific sections of code. A whole foot tap breaks or continues execution, and kicks are used to stop or restart the debugging session.

4.4 Evaluation

We performed small evaluations during our design process, getting fellow lab members or students from the university to try the Tap-Kick-Click system and give qualitative feedback.

Once we settled on a design, we ran a final evaluation with a set of 8 participants recruited from our university campus (4 were female, ages between 20 and 30). As foot input is harder to learn than hand based input [11], the goal of our study was simply to train users to learn to use the system and see if they could perform simple but non-trivial tasks using a real world application. We also asked users to provide feedback on the system in order to identify strengths, weaknesses and areas for improvement of the design. To determine whether the system would be adopted and used for real world tasks by people over the long term would require a much longer study. To support a variety of shoe sizes (which ranged between Women's size 5.5-8 and Men's size 8-9.5), participants could choose between 3 pairs of slippers to find the one of closest size. Only one participant had previously used a standing desk.

The participants then performed a series of tasks designed to introduce them to the system, and evaluate whether they were able to perform non-trivial tasks with a web browser using the system. They used the web browsing mapping from Figure 4-11 with unused commands (add bookmark and Zotero access) removed for simplicity, along with a mapping for simple control of a PDF reader. The experimenter explained components of the system prior to tasks, and provided verbal clarification of the system or task if the user requested it. Participants were allowed to use the keyboard to search queries, but otherwise had to perform the task only with the foot system. The tasks were, in order:

Task 1: Introduce low-density foot actions. Participants performed the foot action (toe tap, kick, etc.) and direction (forward, side, backward) corresponding to a displayed foot action icon (displayed as pairs of foot icons and arrows like those in Figure 4-11). All 22 possible actions were performed. Task 1 was repeated until less than 4 errors were encountered (up to a maximum of 4 times) to ensure a common level of eyes-free performance.

Task 2: Introduce "link mode". Participants navigated through 10 Wikipedia pages, following specified links.

Task 3: Web browsing. Participants selected a pair of Wikipedia pages from a list, and then navigated from the first page to the second page using links. We used the website wikispeedia.net [41], which presents tasks of this form using a version of schools.wikipedia.org.

Task 4: Introduce high-density foot menu. Using a version of the bookmark menu labelled with letters, participants were shown a letter and asked to select the corresponding item using any foot tap. All 20 item locations were selected.

Task 5: Web search and PDF viewing. Participants used the bookmark menu to open the ACM Digital Library, Microsoft Academic Search, and Google Scholar to look up a provided author's citation count. After, they found a specific paper by the author, opened it in a PDF reader, and counted the number of figures in the paper. Finally, they closed the PDF reader and all web browser tabs. (The instructions given to the participant are included in Appendix B)

Tasks 1, 2, and 4 were intended to demonstrate and allow the user to practice using key parts of the system. Tasks 3 and 5 involved using the system to perform a task, requiring the user to simultaneously keep track of how to use the system and the task they were performing.

At the conclusion of the experiment, participants completed a questionnaire (included in Appendix C), which included a simplified version of the NASA-TLX [15] and were asked to provide feedback on the system. Experiment sessions lasted approximately 60 minutes. On average the task portion took 47 minutes (excluding feedback, setup and calibration).

4.4.1 Results

The first goal of our evaluation, to determine if participants could learn and use the system within a study session, was successfully met. All participants were able to successfully complete each task at least once. Although experimenter prompting was required, particularly in the early tasks, participants were generally able to recover from errors (entering the wrong command) using the system, without resorting to the keyboard and mouse. This is encouraging given the complexity and novelty of the system they had to master in a short amount of time. Encouragingly, 6 out of 8 participants were able to complete Task 1 with 3 errors or less on their 3rd repetition of the task.

The secondary focus of the study, to identify strengths, weaknesses and areas for potential improvement, yielded useful feedback on several facets of the design.

We used the numerical questionnaire results to supplement and elicit qualitative feedback from participants, although they were not intended to provide statistically meaningful comparisons.

Factor	Average	Stdev
Mental Demand	8.2	5.0
Physical Demand	8.3	4.2
Temporal Demand	8.2	3.4
Performance	11.8	4.2
Effort	10.1	2.1
Frustration	8	4.6

Table 4-1: NASA-TLX averages across participants
(on a 21 point scale from 0-20, where 0 was best and 20 was worst for each category)

While individual participants reported the task as either more mentally demanding than physically demanding or vice versa, there was no overall consensus on which demand was greater.

Feature	Average	Stdev
Navigation Controls	2.5	0.8
Link Selection	2.5	0.9
Bookmark Menu	3.4	0.7
Application Switching	2	0.8
Feedback Panel	2.4	1.1
Help Screen	1.9	1.1

Table 4-2: Feature ratings across participants
Question: For each of the following features of the interaction technique, please rate on a scale of 5 how easy it was to learn and use.
Scale from 1 (Very Easy) to 5 (Very Hard)

From the questionnaire data it appears that participants found the Bookmark Menu hardest to use, and this result is backed up by qualitative feedback as 6 out of 8 participants reported difficulty using it. On the questionnaire asking about whether the participant had any difficulty using part of the interface, P4 wrote: "Bookmark menu: easy to select wrong [item] because they are very close to each other." Some of this difficulty might be due to lack of practice with the Bookmark Menu as compared to the eyes-free input technique. We initially assumed that the Bookmark Menu would be easy to use without much training as it forces the user to use indirect feedback with a foot cursor, as the menu pops up in the center of the screen. While we only ran the Bookmark Menu training task once, user performance was similar to the first run of the eyes-free training task (40% on Bookmark Menu vs 36% on eyes-free), and running the eyes-free training task multiple times reduced error rate. This suggests that the Bookmark Menu might have better performance once the user has had a chance to practice with it, but this removes the initial justification for using it as a menu selection technique. It might be preferable to simply use the eyes-free input technique for menu selection, with a sequence of commands provided for each item as in the click mode used for link selections. It is also possible that difficulties with the menu are due to sensing errors which are not problematic for the large targets of the eyes-free input technique.

The help menu is reported as being one of the easiest features to use – however, none of participants used it during any of the tasks for its intended purpose, which was to remind the users of difficult commands. Thus, it seems that this result reflects the ease of use of the two-foot jumping command, but the help menu itself is not very useful.

Four participants reported that having to look commands up on the sidebar created additional cognitive load while performing the tasks. However, 3 out of 4 also felt that this would get easier with practice. P8 summed this up when explaining a high ranking on the NASA-TLX for Mental Demand, saying: “[It is mentally demanding] in the sense that you have to think a lot; it was easy to get mixed up between the different commands sometimes. Until you get the hang of it - once I got the hang of it, it seems pretty easy.” This suggests that there is room for additional support in teaching commands to new users, perhaps in the form of an explicit tutorial, or a training task which prompts the user with a command (go back in the web browser) and requires the user to perform the correct foot action. We also asked participants whether the task would be appropriate for 5 of our suggested application scenarios. Participants supported the use of the system for Controlling a Web Browser (7), Controlling a Background Application (8), and Facebook Blocking (7), were lukewarm about Controlling a Citation Manager (5) and generally opposed to using it to the Controlling a Debugger scenario (only 2 in favour). We did not ask for participant’s level of programming experience which may have influenced results, but one participant expressed that they thought using the system would add to the cognitive difficulty of debugging a program.

Three participants wanted to use a larger interaction area or encountered sensing errors caused by kicking too far. The interaction area was calibrated based on a conservative estimate of interaction range from the previous study, but this suggests that it may be preferable to allow the user to calibrate their own interaction range.

Link selection was reported as moderately easy to use in the questionnaire, and participants were able to perform it multiple times to complete tasks 2, 3 and 5. Two participants complained about the link annotations covering up the text of the links, and one participant said that link selection had high Physical Demand because of the number of actions required.

4.4.1.1 Summary of Results

To summarize, our main finding from the user evaluation was that participants were able to successfully learn to use our system and perform tasks in a web browser within a 60 minute time frame. When they made mistakes, they were usually able to recover within the framework of the system (though experimenter prompting was still sometimes required).

We also received useful feedback on several aspects of the design:

- Navigation controls, link selection, application switching and feedback panel were generally well received, although with somewhat of a learning curve.
- The bookmark menu in its current form is somewhat difficult to learn and use – it may be better to use the same click mode as used in link selection to simplify learning.
- The help screen is currently not used by participants. It might need to be reworked into a more relevant form, such as a tutorial or training task for the different web browser commands.
- Some minor system changes would improve performance, including increasing the interaction range and making toe and heel tap detection more conservative to improve performance of whole foot tapping detection.

Chapter 5

Conclusion

In this chapter, we present a summary of the design process described in this thesis, comprising the motivation for the work (Chapter 1), survey of previous research into foot input (Chapter 2), the empirical studies we conducted to understand performance characteristics of foot input (Chapter 3) and the design and evaluation of Tap-Kick-Click (Chapter 4). We also describe areas that could be fruitful for additional study, either by extending the system we created or performing additional experiments to better understand how users interact with it.

5.1 Future Work

There are several potential directions for future research arising from the work described in this thesis:

- A study could be designed in the Fitts's Law paradigm [24] to examine whether something analogous to Fitts's Law can be derived for 2d foot tapping in a standing context. It would need to differ from our quantitative studies in looking at a more values of a smaller number of variables.
- A long term study of the Tap-Kick-Click system and interaction technique could be performed with workers who use a standing desk on a regular basis. This would allow the system to be tested with real world tasks, provide an opportunity for users to reach expert level performance with the system, allow for health benefits of the system to be investigated, and determine whether users will adopt the system in the long run.
- It would be possible to extend the Tap-Kick-Click interaction technique to a mobile context, by replacing the camera based sensing with wearable accelerometer sensing. Several challenges would need to be addressed, including training, implementing an appropriate delimiter for interaction, and selecting an appropriate level of feedback (visible feedback could be provided via a head mounted display such as Google Glass, while eyes-free feedback could be provided using vibration motors).
- There are opportunities to extend Tap-Kick-Click with additional interaction methods for specific tasks. If routinely navigating large information spaces, our vocabulary could be

extended with continuous rate-control methods like Alexander et al. [1]. Additional natural postures could be used for specific tasks, for example crossing one foot behind the other to switch between two virtual desktops. Or different whole body movements, such as arm movements, could be combined with the foot movements to increase command throughput.

- We briefly propose combining speech recognition with foot input, to provide text input and allow for correction of recognition errors. This area could be investigated further.

5.2 Conclusion

We began the work described in this thesis with a goal in mind: to create a system for foot interaction that computer workers could use to take healthy breaks while performing work with regular desktop applications (Chapter 1). We surveyed the literature (Chapter 2) for foot input systems that would be suited for this context, but found that although foot input had been explored for seated input, mobile interaction and immersive environments, little similar research had been previously conducted. We found the work-in-progress by Meyers et al. [27] describing their concept of Step User Interfaces, and showing examples of how this could be applied to simple tasks. However, their work was not directly scalable to more complex interaction, and did not include empirical evaluation of input performance. We examined the choices made by other foot input work in sensing and in the type of foot actions they used for interaction. We also surveyed previous empirical studies of foot performance and design guidelines proposed by previous work. We settled on using a interaction technique (Section 3.1) involving discrete indirect foot pointing with kicking and tapping to provide input suitable for controlling applications, to allow flexibility in the configuration of targets mapped to commands, and to allow the user to keep their attention on a screen instead of on their feet.

But to implement this technique, we needed to answer a number of questions about empirical performance (Chapter 3), including: what target configuration to use, in terms of size and location; whether user preference or performance characteristics would rule out using tapping or kicking in interaction; and how much of the user’s attention would need to be devoted to monitoring the sensing system. To answer these questions, we carried out two empirical experiments on foot performance, prototyping the interaction using a Vicon motion capture system. The first experiment focused on deriving guidelines on target size and location for indirect foot tapping. We found that targets of approximately 10 cm radial size and 22° provided reasonable time and error rates, that foot dominance had little impact, and that tapping was more difficult in the backwards direction. The second experiment built on these results to compare tapping and kicking directly, and to determine whether the user could successfully tap on targets without a visible foot cursor. We found that both tapping and kicking were useable, but tapping was preferred; and that interaction was much more difficult without a foot cursor. We combined the results of both studies into a set of design guidelines (Section 3.4.2).

With these guidelines in hand, we were able to instantiate the interaction technique in Tap-Kick-Click, a complete system for performing tasks in desktop applications using foot input (Chapter 4). We implemented a custom computer vision algorithm for the Microsoft Kinect that tracked an IR LED attached to a pair of slippers, and augmented the slippers with a set of pressure sensors. The system combined kicking and 3 types of toe tapping with a set of 6 large targets to create a set of actions that could be mapped to application commands. Using large targets and providing feedback in an always visible sidebar, we approximated an eyes-free input technique which would allow a user to troubleshoot misrecognized commands. To make learning the mapping between foot actions and application commands easy, we provided a concise representation of foot action and application commands in the sidebar, and provided a help screen activated by a two foot jump. To control a web browser, we implemented Click Mode, where links on a webpage were labeled with a sequence of arrows indicating targets to tap on to select the link. To assist in selecting items from a menu, we implemented a denser target layout. And to help users reduce wasted time on websites like Facebook, we implemented an anti-cyberslacking mode which would force a user to assume a mildly uncomfortable posture to continue using the website.

Once implementation was complete, we ran a user evaluation of the system (Section 4.4). We found some potential areas for future improvement of the system as the bookmark menu was somewhat difficult to use, the help screen was not used by participants when they got stuck, and some aspects of the sensing system could be tweaked. But the main result of the study validated our approach by finding that users were able to successfully learn the system and perform tasks in a web browser within the space of an hour long experimental session. In the end, our work reached its intended conclusion of providing a small step towards healthier computer interaction for desk-bound computer workers.

Appendix A

Questionnaire used in Experiment 2

Circle the most appropriate answer for each question:

For TAP WITH FEEDBACK:

1. It is easy to perform mentally
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

2. It is easy to perform physically
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

3. It is easy to learn
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

For KICK WITH FEEDBACK:

1. It is easy to perform mentally
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

2. It is easy to perform physically
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

3. It is easy to learn
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

For KICK WITH FEEDBACK and TAP WITH FEEDBACK:

1. Between KICK WITH FEEDBACK and TAP WITH FEEDBACK, which do you prefer?
(1) Strongly Prefer KICK WITH FEEDBACK
(2) Slightly Prefer KICK WITH FEEDBACK
(3) Have No Preference
(4) Slightly Prefer TAP WITH FEEDBACK
(5) Strongly Prefer TAP WITH FEEDBACK

2. Why do you have this preference?

For TAP WITHOUT FEEDBACK:

1. It is easy to perform mentally
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

2. It is easy to perform physically
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

3. It is easy to learn
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

For KICK WITHOUT FEEDBACK:

1. It is easy to perform mentally
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

2. It is easy to perform physically
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

3. It is easy to learn
(1) Strongly Agree (2) Agree (3) Neither (4) Disagree (5) Strongly Disagree

For KICK WITHOUT FEEDBACK and TAP WITHOUT FEEDBACK:

1. Between KICK WITHOUT FEEDBACK and TAP WITHOUT FEEDBACK, which do you prefer?
(1) Strongly Prefer KICK WITHOUT FEEDBACK
(2) Slightly Prefer KICK WITHOUT FEEDBACK
(3) Have No Preference
(4) Slightly Prefer TAP WITHOUT FEEDBACK
(5) Strongly Prefer TAP WITHOUT FEEDBACK

2. Why do you have this preference?

For all techniques:

Did you experience any physical discomfort while performing the tasks in this experiment? If so, please describe what you experienced, what part of the task you think caused it, and whether it affected your ability to perform the task.

Did you notice any differences in performing the tasks in this experiment based on target direction, size or distance?

Did you try any variations on the techniques described by the experimenter? Were they effective?

Do you think that the techniques involved in the experiment could be used as part of a computer interface (for example, to perform actions in a web browser, such as scrolling, selecting links and switching tabs)?

Do you have any additional comments?

Appendix B

Task 5 from System Evaluation

1. Use the bookmark Menu to open the link “List of Authors”
2. On the page is a list of authors. For the first author listed, go through the following instructions:
   a) Open each of the following 3 websites in a separate tab using the Bookmark Menu, and search them for the name of the author:
      i. ACM Digital Library
      ii. Microsoft Academic Search
      iii. Google Scholar

      On each of the websites, find and report the citation count listed for the author.
      You may have to find a link to a specific author profile page to get these results – if the author’s name shows up as a link in the search results, try following it.
   b) Open the google scholar user profile page and locate the first paper published at the SIGCHI conference. Follow links until you can open a pdf version of the paper (if the paper does not have a pdf version available, select the next one).
   c) Count the number of figures by scrolling through the paper
   d) Close the pdf window
   e) Close the all of the extra tabs you opened (leaving open the List of Authors page)

Appendix C

Questionnaire used in System Evaluation

For each of the following features of the interaction technique, please rate on a scale of 5 how easy it was to learn and use:

Navigation Controls
(Controlling the web browser by moving forward/back, switching tabs, etc.)
(1) Very Easy (2) Easy (3) Neutral (4) Hard (5) Very Hard

Link Selection
(Selecting links to follow on a web page)
(1) Very Easy (2) Easy (3) Neutral (4) Hard (5) Very Hard

Bookmark Menu
(The menu of frequently used websites)
(1) Very Easy (2) Easy (3) Neutral (4) Hard (5) Very Hard

Application Switching
(Switching between different applications)
(1) Very Easy (2) Easy (3) Neutral (4) Hard (5) Very Hard

Feedback Panel
(The panel at the side of the page, indicating which commands are currently available, and which command is currently active)
(1) Very Easy (2) Easy (3) Neutral (4) Hard (5) Very Hard

Help Screen
(The window, activated by a two foot jump, that displayed the currently available commands)
(1) Very Easy (2) Easy (3) Neutral (4) Hard (5) Very Hard

Were there any parts of the task or interface that you had difficulty with?

Do you have any suggestions for improvement of the interaction technique?

Do you have any preference regarding the assignment of foot actions to commands? Was there one kind of foot action you preferred or disliked?

Consider the following application scenarios. Lots of people use a standing desk for health and the goal of our system is to support this. Imagine you are a person interested in health using a standing desk. For each of the following scenarios, would you consider using the foot input system from this experiment to perform the described task?

1. Control a web browser
(Use the foot controls to navigating through web pages, select links, etc.)
I would use the system for this purpose: Yes No

2. Control a background application
(While working on another task, using the foot controls to change tracks, change volume or play/pause in a music player)
I would use the system for this purpose: Yes No

3. Facebook blocking
(When you open up a distracting website like Facebook, you are required to assume an uncomfortable position to continue using the website, to encourage you to waste less time)
I would use the system for this purpose: Yes No

4. Control a debugger
(While debugging a program in an application like Visual Studio, using the foot controls to resume/break the application and step into or out of specific section of code)
I would use the system for this purpose: Yes No

5. Control a citation manager
(Use to foot controls to navigate through a citation manager, select web sites or pdfs to open, and switch between the citation manager, a web browser, and a pdf reader)
I would use the system for this purpose: Yes No

The scenario (out of the above) that would be best for this system is ________

The scenario (out of the above) that would be worst for this system is ________

Do you think that the interaction technique involved in the experiment could be used for any additional purpose(s)?

Do you have any additional comments?

Bibliography

1. Alexander, J., Han, T., Judd, W., Irani, P., and Subramanian, S. Putting Your Best Foot Forward: Investigating Real-world Mappings for Foot-based Gestures. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2012), 1229–1238.

2. Andersen, L.L., Mortensen, O.S., Hansen, J.V., and Burr, H. A prospective cohort study on severe pain as a risk factor for long-term sickness absence in blue- and white-collar workers. *Occupational and Environmental Medicine 68*, 8 (2011), 590–592.

3. Augsten, T., Kaefer, K., Meusel, R., et al. Multitoe: high-precision interaction with back-projected floors based on high-resolution multi-touch input. *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, ACM (2010), 209–218.

4. Berque, D., Burgess, J., Billingsley, A., Johnson, S., Bonebright, T.L., and Wethington, B. Design and evaluation of persuasive technology to encourage healthier typing behaviors. *Proceedings of the 6th International Conference on Persuasive Technology: Persuasive Technology and Design: Enhancing Sustainability and Health*, ACM (2011), 9:1–9:10.

5. Carrozzino, M., Avveduto, G., Tecchia, F., Gurevich, P., and Cohen, B. Navigating Immersive Virtual Environments Through a Foot Controller. *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology*, ACM (2014), 23–26.

6. Chau, J.Y., Grunseit, A.C., Chey, T., et al. Daily Sitting Time and All-Cause Mortality: A Meta-Analysis. *PLoS ONE 8*, 11 (2013), e80000.

7. Crossan, A., Brewster, S., and Ng, A. Foot tapping for mobile interaction. *Proceedings of the 24th BCS Interaction Specialist Group Conference*, (2010), 418–422.

8. Dearman, D., Karlson, A., Meyers, B., and Bederson, B. Multi-modal text entry and selection on a mobile device. *Proceedings of Graphics Interface 2010*, Canadian Information Processing Society (2010), 19–26.

9. Dixon, M. and Fogarty, J. Prefab: implementing advanced behaviors using pixel-based reverse engineering of interface structure. *Proc. of CHI*, (2010), 1525–1534.

10. Drury, C.G. Application of Fitts' Law to foot-pedal design. *Human Factors: The Journal of the Human Factors and Ergonomics Society 17*, 4 (1975), 368–373.

11. Garcia, F.P. and Vu, K.-P.L. Effectiveness of hand- and foot-operated secondary input devices for word-processing tasks before and after training. *Computers in Human Behavior 27*, 1 (2011), 285–295.

12. Göbel, F., Klamka, K., Siegel, A., Vogt, S., Stellmach, S., and Dachselt, R. Gaze-supported foot interaction in zoomable information spaces. *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, ACM (2013), 3059–3062.

13. Grunseit, A.C., Chau, J.Y.-Y., van der Ploeg, H.P., and Bauman, A. "Thinking on your feet": A qualitative evaluation of sit-stand desks in an Australian workplace. *BMC Public Health 13*, (2013), 365.

14. Han, T., Alexander, J., Karnik, A., Irani, P., and Subramanian, S. Kick: investigating the use of kick gestures for mobile interactions. *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, (2011), 29–32.

15. Hart, S.G. and Staveland, L.E. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In P.A.H. and N. Meshkati, ed., *Advances in Psychology*. North-Holland, 1988, 139–183.

16. Henning, R.A., Jacques, P., Kissel, G.V., Sullivan, A.B., and Alteras-Webb, S.M. Frequent short rest breaks from computer work: effects on productivity and well-being at two field sites. *Ergonomics 40*, 1 (1997), 78–91.

17. Hoffmann, E.R. A comparison of hand and foot movement times. *Ergonomics 34*, 4 (1991), 397–406.

18. Hoysniemi, J. International survey on the Dance Dance Revolution game. *Comput. Entertain. 4*, 2 (2006).

19. Jalaliniya, S., Smith, J., Sousa, M., Büthe, L., and Pederson, T. Touch-less Interaction with Medical Images Using Hand & Foot Gestures. *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, ACM (2013), 1265–1274.

20. Jota, R., Lopes, P., Wigdor, D., and Jorge, J. Let's Kick It: How to Stop Wasting the Bottom Third of Your Large Screen Display. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2014), 1411–1414.

21. Kim, S.-H. and Kaber, D.B. Design and Usability Evaluation of Foot Interfaces for Dynamic Text Editing. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting 52,* 6 (2008), 581–585.

22. Larsson, B., Søgaard, K., and Rosendal, L. Work related neck–shoulder pain: a review on magnitude, risk factors, biochemical characteristics, clinical picture and preventive interventions. *Best Practice & Research Clinical Rheumatology 21*, 3 (2007), 447–463.

23. Lv, Z., Halawani, A., Feng, S., Li, H., and Réhman, S.U. Multimodal Hand and Foot Gesture Interaction for Handheld Devices. *ACM Trans. Multimedia Comput. Commun. Appl. 11*, 1s (2014), 10:1–10:19.

24. MacKenzie, I.S. Fitts' law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact. 7*, 1 (1992), 91–139.

25. Matthies, D.J., Müller, F., Anthes, C., and Kranzlmüller, D. ShoeSoleSense: proof of concept for a wearable foot interface for virtual and real environments. *Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology*, (2013), 93–96.

26. Mclean, L., Tingley, M., Scott, R.N., and Rickards, J. Computer terminal work and the benefit of microbreaks. *Applied ergonomics 32*, 3 (2001), 225–237.

27. Meyers, B., Brush, A.J.B., Drucker, S., Smith, M.A., and Czerwinski, M. Dance your work away: exploring step user interfaces. *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, ACM (2006), 387–392.

28. Morris, D., Brush, A.J.B., and Meyers, B.R. SuperBreak: Using Interactivity to Enhance Ergonomic Typing Breaks. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2008), 1817–1826.

29. Paelke, V., Reimann, C., and Stichling, D. Foot-based mobile interaction with games. *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology*, ACM (2004), 321–324.

30. Pakkanen, T. and Raisamo, R. Appropriateness of foot interaction for non-accurate spatial tasks. *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, ACM (2004), 1123–1126.

31. Paradiso, J.A. and Hu, E. Expressive Footwear for Computer-Augmented Dance Performance. *Proceedings of the 1st IEEE International Symposium on Wearable Computers*, IEEE Computer Society (1997), 165–.

32. Pearson, G. and Weiser, M. Of Moles and Men: The Design of Foot Controls for Workstations. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (1986), 333–339.

33. Pearson, G. and Weiser, M. EXPLORATORY EVALUATIONS OF TWO VERSIONS OF A FOOT-OPERATED CURSOR-POSITIONING DEVICE IN A TARGET-SELECTION TASK. *SIGCHI Bull. 19*, 3 (1988), 70–75.

34. Pedrosa, D. and Pimentel, M. da G.C. Text Entry Using a Foot for Severely Motor-impaired Individuals. *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, ACM (2014), 957–963.

35. Rico, J. and Brewster, S. Usable gestures for mobile interfaces: evaluating social acceptability. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2010), 887–896.

36. Schmidt, D., Block, F., and Gellersen, H. A comparison of direct and indirect multi-touch input for large surfaces. In *Human-Computer Interaction–INTERACT 2009*. Springer, 2009, 582–594.

37. Schmidt, D., Ramakers, R., Pedersen, E.W., et al. Kickables: Tangibles for Feet. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2014), 3143–3152.

38. Schöning, J., Daiber, F., Krüger, A., and Rohs, M. Using hands and feet to navigate and manipulate spatial data. *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, ACM (2009), 4663–4668.

39. Scott, J., Dearman, D., Yatani, K., and Truong, K.N. Sensing foot gestures from the pocket. *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, ACM (2010), 199–208.

40. Straker, L., Levine, J., and Campbell, A. The Effects of Walking and Cycling Computer Workstations on Keyboard and Mouse Performance. *Human Factors: The Journal of the Human Factors and Ergonomics Society 51*, 6 (2009), 831–844.

41. West, R., Pineau, J., and Precup, D. Wikispeedia: An Online Game for Inferring Semantic Distances between Concepts. *IJCAI*, (2009), 1598–1603.

42. Whitehead, A., Johnston, H., Nixon, N., and Welch, J. Exergame effectiveness: what the numbers can tell us. *Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games*, ACM (2010), 55–62.

43. Wing, A.M. Type Writer Carriage Return. 1915. http://www.google.ca/patents?id=KhxrAAAAEBAJ.

44. Wu, L. Slider-based Foot Input Devices. 2010. http://graphics.stanford.edu/~lwu2/papers/SliderMouse2010.pdf.

45. Yamamoto, T., Tsukamoto, M., and Yoshihisa, T. Foot-Step Input Method for Operating Information Devices While Jogging. *Proceedings of the 2008 International Symposium on Applications and the Internet*, IEEE Computer Society (2008), 173–176.

46. Rise Up Against Sitting Disease: 22 Healthy Ways to Move More | Reader's Digest. http://www.rd.com/health/healthcare/rise-up-against-sitting-disease/.

47. Sitting too long is making us sick: How to combat sitting disease. http://www.cbc.ca/thecurrent/episode/2013/09/24/preventing-sitting-disease-by-standing-at-work/.

48. Doug Engelbart Discusses Mouse Alternatives. ftp://ftp.cs.utk.edu/pub/shuford/terminal/engelbart_mouse_alternatives.html.

49. FOOTLOGGER | FootLogger. http://footlogger.com:8080/hp_new/footlogger/.

50. Home - Sensoria Fitness. http://www.sensoriafitness.com/.

51. AutoHotkey: macro and automation Windows scripting language. http://www.autohotkey.com/.

52. Hit-a-Hint. https://addons.mozilla.org/en-US/firefox/addon/hit-a-hint/.